

HI-REASO: A Multi-Level Multi-Faceted Hindi Mathematical Evaluation Suite for Foundation Models

Md Tanzib Hosain¹, Rajan Das Gupta², Zannatul Fardaush Tripty³, Md Kishor Morol⁴

¹MPI-SP, ²Jahangirnagar University,

³Chittagong University of Engineering and Technology, ⁴Meta FAIR

tanzib.hosain@mpi-sp.org, rdasgupta@juniv.edu,

u1804030@student.cuet.ac.bd, morolkishor@meta.com

Abstract

Large Language Models (LLMs) have shown impressive capabilities across many language understanding tasks; however, their ability to perform numerical reasoning remains fragile, particularly outside English. This limitation is especially concerning for Hindi, a language with hundreds of millions of speakers but relatively little focused evaluation in numerically grounded reasoning. In this work, we present HI-REASO, a benchmark designed to systematically evaluate numerical reasoning in Hindi. The benchmark comprises six complementary task types and 3k instances drawn from educational materials, real-world problem formulations, and commonsense reasoning scenarios. We assess a range of state-of-the-art LLMs using Hindi-native prompting as well as cross-lingual prompting and cross-lingual chain-of-thought reasoning. Across all settings, models consistently underperform compared to human baselines, with the largest performance degradation observed when reasoning is conducted directly in Hindi. While cross-lingual strategies offer partial improvements, they do not close the gap. Our findings reveal fundamental shortcomings in current LLMs’ numerical reasoning abilities under Hindi-native conditions and motivate the development of models and training paradigms that better support linguistically diverse numerical reasoning. We publicly release the code and data at <https://github.com/kraritt/hi-reaso>.

1 Introduction

Numerical reasoning is a core capability required for many real-world applications, including education, finance, and scientific problem solving. In natural language, numerical information is embedded in text, requiring models to identify quantities, interpret relations, and map linguistic descriptions to formal mathematical representations.

Large Language Models (LLMs) have achieved strong performance across diverse natural language processing tasks (Brown et al., 2020; Zhao et al., 2026), yet numerical reasoning remains a persistent challenge. Prior work shows that LLMs are sensitive to superficial variations in problem formulations and frequently fail on multi-step numerical reasoning tasks (Bang, 2023; Patel et al., 2021; Ahn et al., 2024). While techniques such as chain-of-thought prompting improve performance by eliciting intermediate reasoning steps (Wei et al., 2022), these gains are limited, particularly in multilingual and non-English settings (Shi et al., 2022a; Qin et al., 2023; Huang et al., 2023).

Several benchmarks have been proposed to evaluate numerical reasoning. English-centric datasets such as NumGLUE and LILA assess a range of numerical reasoning skills (Mishra et al., 2022b,a), while multilingual studies highlight substantial variation in reasoning performance across languages (Shi et al., 2022a; Qin et al., 2023). However, these benchmarks provide limited insight into language-specific numerical reasoning challenges.

This limitation is particularly evident for Hindi, one of the most widely spoken languages worldwide. Although Indic benchmarks such as IndicGLUE and IndicXNLI support general language understanding evaluation (Sankalp et al., 2025; Aggarwal et al., 2022; Khanuja et al., 2020), they do not explicitly target numerical reasoning, leaving Hindi numerical reasoning largely underexplored.

In this work, we introduce HI-REASO, a benchmark designed to systematically evaluate numerical reasoning in Hindi. The benchmark comprises six task categories and 3k instances curated from educational materials, real-world problem descriptions, and commonsense numerical scenarios. We evaluate state-of-the-art LLMs using Hindi Native Prompting (HNaP), Cross-Lingual Prompting (XLP), and Cross-Lingual Chain-of-Thought

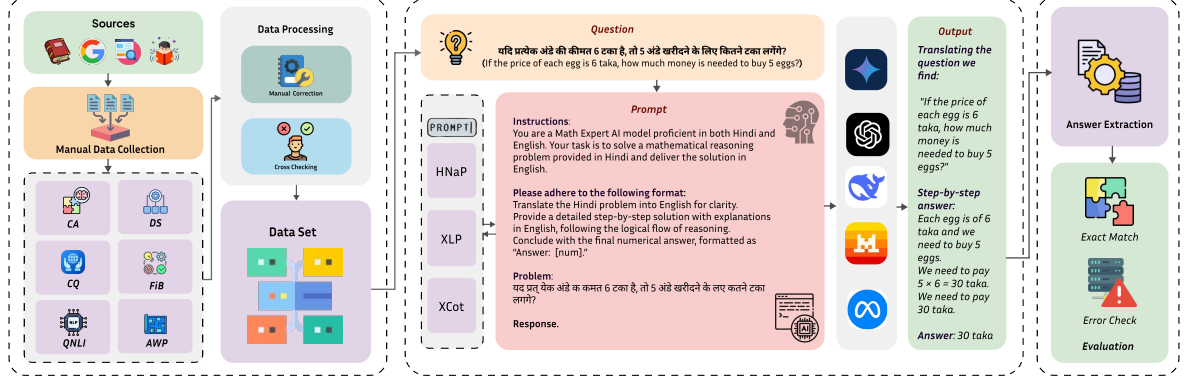


Figure 1: Overview of HI-REASO. The figure shows the dataset construction pipeline, the experimental setup with different numerical reasoning prompts, and a system example where a Hindi numerical question is combined with the XCoT prompt to form a language-independent input for step-by-step reasoning and evaluation by an LLM. .

Prompting (XCoT), enabling a diagnostic comparison of language-dependent reasoning performance.

Our results show that LLMs consistently underperform human baselines across all tasks, with the largest performance degradation observed under Hindi-native prompting. Although cross-lingual strategies yield partial improvements, they fail to close the gap, highlighting persistent limitations in current models’ numerical reasoning abilities.

The main contributions of this work are:

- We introduce HI-REASO, the first benchmark for evaluating numerical reasoning in Hindi, covering six task types with 3k curated instances.
- We present a systematic evaluation of LLMs under Hindi-native and cross-lingual prompting settings, providing diagnostic insights into language-dependent reasoning failures.
- We demonstrate a substantial gap between model and human performance, motivating future research on language-aware numerical reasoning.

2 Overview of Tasks in HI-REASO

Evaluating numerical reasoning in Large Language Models (LLMs) is particularly challenging in non-English settings due to limited linguistic resources and differences in numerical expression. To address this challenge for Hindi, we introduce HI-REASO, a benchmark comprising six task categories focused on evaluating numerical reasoning in Hindi (see Table 1). The tasks span arithmetic

computation, commonsense reasoning, domain-specific knowledge, and quantitative inference.

Commonsense Arithmetic (CA). Consider the problem: अभय ने बोर्ड पर हर हिंदी स्वर लिखा। यदि प्रत्येक स्वर 7 बार लिखा गया, तो बोर्ड पर कुल कितने अक्षर लिखे गए? (*Abhay wrote each Hindi vowel on the board. If each vowel was written 7 times, how many letters were written on the board in total?*) Solving this problem requires both arithmetic reasoning and numerical commonsense knowledge, such as recognizing the number of Hindi vowels. These tasks combine common numerical facts (e.g., days in a week, fingers on a hand) with arithmetic computation.

Domain-Specific (DS). 12 रेडियन को डिग्री में व्यक्त करने पर मान कितना होगा? (*What is the value of 12 radians in degrees?*) This task requires numerical reasoning combined with domain-specific knowledge, such as applying the formula $\text{Degrees} = \frac{180}{\pi} \times \text{Radians}$. The dataset includes problems from domains such as physics, chemistry, advanced mathematics, and computer science.

Commonsense Quantitative (CQ). This category involves quantitative comparison tasks. For example: संख्या 1234567890 क्या 9876543210 से छोटी है या बड़ी? (1) छोटी (2) बड़ी (*Is the number 1234567890 smaller or larger than 9876543210? (1) Smaller (2) Larger*) These questions may require direct magnitude comparison or intermediate arithmetic operations.

Fill-in-the-Blanks (FiB). 4 पेन की कीमत 80 रुपये है। 10 पेन की कीमत _____ रुपये होगी। (*The price of 4 pens is 80 rupees. The price of 10 pens is _____*)

Table 1: Task categories in HI-REASO, dataset, and representative examples. Hindi examples are shown first, followed by English glosses.

Tasks	Examples (Hindi → English)
CA	Q: यदि 5 दर्जन अंडों की कीमत 300 रुपये है, तो 7 दर्जन अंडों की कीमत कितनी होगी? (<i>If the price of five dozen eggs is 300 rupees, what is the price of seven dozen eggs?</i>) A: 420 रुपये (420 rupees)
DS	Q: एक बाइक 7 सेकंड में 21 मीटर चलती है। बाइक की औसत गति ज्ञात कीजिए। (<i>A bike travels 21 meters in 7 seconds. Calculate the average speed of the bike.</i>) A: 3 मीटर/सेकंड (3 meters/second)
CQ	Q: एक पेड़ की शाखा की मोटाई 6 इंच है और तने की मोटाई 0.25 इंच है। कौन अधिक मोटा है? (<i>The branch is 6 inches thick, while the trunk is 0.25 inches thick. Which is thicker?</i>) A: शाखा (Branch)
FiB	Q: तीन संख्याओं का योग 5,452 है। संख्याएँ 2,563, ____, और 1,245 हैं। (<i>The sum of three numbers is 5,452. The numbers are 2,563, ____, and 1,245.</i>) A: 1,644 (1,644)
QNLI	Premise: राहुल ने 65 किताबें खरीदीं। (<i>Rahul bought 65 books.</i>) Hypothesis: राहुल ने 25 से अधिक किताबें खरीदीं। (<i>Rahul bought more than 25 books.</i>) Label: Entailment
AWP	Q: यदि 8 किलोग्राम चावल की कीमत 960 रुपये है, तो 4,800 रुपये में कितने किलोग्राम चावल खरीदे जा सकते हैं? (<i>If the price of 8 kg of rice is 960 rupees, how many kg can be bought for 4,800 rupees?</i>) A: 40 किलोग्राम (40 kg)

rupees.) Unlike tasks requiring external knowledge, this category is self-contained and focuses solely on numerical reasoning, following a fill-in-the-blank formulation.

Quantitative Natural Language Inference (QNLI). पूर्वकथन: राहुल ने 50 रुपये में एक किताब खरीदी। परिकल्पना: राहुल ने 1 रुपये में 80 किताबें खरीदीं। (*Premise: Rahul bought a book for 50 rupees. Hypothesis: Rahul bought 80 books for one rupee.*) This task evaluates the relationship between a premise and a hypothesis using numerical reasoning, framing arithmetic problems as natural language inference tasks.

Arithmetic Word Problems (AWP). Arithmetic word problems require interpreting real-world scenarios and performing arithmetic calculations. For example: 15 बॉल पेन 75 रुपये में खरीदे गए और 90 रुपये में बेच दिए गए। प्रतिशत लाभ कितना होगा? (*If 15 ballpoint pens are bought for 75 rupees and sold for 90 rupees, what is the percentage of profit?*) This task assesses the ability to apply arithmetic reasoning in practical, real-world contexts.

3 Dataset Development

Constructing a benchmark for numerical reasoning is inherently challenging due to the sensitiv-

ity of numerical tasks to linguistic variation and annotation errors. These challenges are amplified in low-resource, non-English settings such as Hindi, where high-quality, domain-specific resources are limited. Additional difficulties include curating diverse problem types across multiple task formats and ensuring mathematical correctness through careful annotation. The dataset development pipeline for HI-REASO is illustrated in Figure 1.

3.1 Data Collection

Manual Data Collection. Arithmetic and word-problem instances were manually sourced from Hindi-medium school textbooks¹ aligned with national and state-level curricula, including materials published by NCERT² and state education boards³. For the Domain-Specific (DS) task, problems were collected from standard academic books covering Physics, Chemistry, and Higher Mathematics. These sources ensure linguistic naturalness and curriculum relevance.

We observed that certain task categories, specifically Commonsense Quantitative (CQ) and Quantitative Natural Language Inference (QNLI), were

¹<https://epathshala.nic.in/>

²<https://www.ncert.nic.in/>

³<https://www.cbse.gov.in/>

underrepresented in school textbooks. To address this gap, relevant problems were adapted from existing numerical reasoning benchmarks and translated into Hindi using automated translation tools⁴. All translated instances were manually reviewed to correct mistranslations, incorrect numerals, and residual non-Hindi artifacts.

Commonsense Arithmetic (CA) problems were additionally curated using widely accepted numerical commonsense knowledge encountered in everyday contexts. Where appropriate, culturally grounded Hindi contexts were incorporated to improve the naturalness of the problems.

Task Formatting. Collected problems were reformatted into task-specific templates corresponding to the six evaluation categories. For example, selected arithmetic word problems were converted into fill-in-the-blank formats to support the FiB task, while preserving the original numerical reasoning requirements.

3.2 Data Processing

Cross-Checking. The dataset was derived from two primary sources: Hindi-medium educational textbooks (for CA, DS, FiB, and AWP tasks) and adapted benchmark problems (for CQ and QNLI tasks). Since these sources had been previously validated by educators or prior benchmark efforts, recomputation of answers was generally unnecessary. Nevertheless, all instances were cross-checked to identify typographical or numerical inconsistencies introduced during collection or translation, and duplicate entries were removed.

Manual Validation. Minor annotation errors were identified in fewer than 1% of the instances. These cases were manually reviewed and corrected by an experienced high school science teacher, improving the dataset’s overall quality, consistency, and reliability.

4 Dataset Distributions

The HI-REASO dataset is designed to cover a broad spectrum of numerical reasoning skills. This section presents an overview of the dataset distribution, analyzing its composition across task categories and data sources. We examine the balance of task types, sample sizes, and problem formats, highlighting how the dataset systematically chal-

lenges models across multiple dimensions of numerical reasoning.

Table 2: Statistics of the HI-REASO dataset grouped by task categories and data sources.

Group	#Instances (%)
<i>Grouped by Task</i>	
CA	650 (21.59%)
DS	652 (21.66%)
CQ	370 (12.29%)
QNLI	120 (3.99%)
FiB	616 (20.46%)
AWP	602 (20.01%)
<i>Grouped by Source</i>	
School Math Book	1080 (35.88%)
Translated	860 (28.57%)
DS Math	327 (10.86%)
Manual Curation	160 (5.32%)
Physics	280 (9.30%)
Chemistry	230 (7.64%)
CS	73 (2.43%)
Total	3010 (100.00%)

4.1 Task-wise Dataset Distribution

The task-wise distribution of HI-REASO reflects a deliberate focus on diverse numerical reasoning phenomena. Domain-Specific (DS) tasks constitute 21.66% of the dataset, emphasizing contextualized arithmetic grounded in specialized domains and real-world applications. Commonsense Quantitative (CQ) tasks account for 12.29%, integrating numerical comparison with general reasoning, a relatively underexplored but important aspect of numerical understanding.

Fill-in-the-Blanks (FiB) tasks form 20.46% of the dataset, underscoring the importance of fundamental arithmetic skills across varied contexts. Together, the remaining categories, including Commonsense Arithmetic (CA), Quantitative Natural Language Inference (QNLI), and Arithmetic Word Problems (AWP), contribute to a balanced evaluation framework. This distribution enables a holistic assessment of numerical reasoning, spanning both general arithmetic proficiency and domain-specific problem solving (see Table 1 and Table 2).

4.2 Source-wise Dataset Distribution

HI-REASO integrates traditional educational resources with translated and manually curated problems to ensure comprehensive coverage of numerical reasoning scenarios. A substantial portion of the dataset (35.88%) is sourced from Hindi-medium school mathematics textbooks, contribut-

⁴<https://pypi.org/project/deep-translator/>

ing primarily to CA, FiB, and AWP tasks and aligning the dataset with formal educational settings.

To increase linguistic and contextual diversity, 28.57% of the dataset consists of translated problems, which primarily support CQ and QNLI tasks while also contributing to CA. Domain-specific mathematical problems account for 10.86% of the dataset and are dedicated to DS tasks. In addition, subject-specific problems from Physics (9.30%) and Chemistry (7.64%) further enrich the dataset by incorporating real-world scientific contexts that require numerical reasoning. A smaller portion (2.43%) originates from Computer Science, focusing on number systems and contributing to DS tasks.

Finally, 5.32% of the dataset comprises manually curated problems aimed at evaluating commonsense arithmetic reasoning, reflecting everyday numerical scenarios encountered in real-life contexts. By integrating such diverse sources, HI-REASO ensures a balanced representation of formal educational content, domain-specific knowledge, and linguistic diversity in Hindi. This composition establishes it as a robust benchmark for assessing numerical reasoning in low-resource language settings (see Table 2). A comprehensive analysis of dataset diversity is provided in Appendix B, and task-wise difficulty characterization is detailed in Appendix A.

5 Experiments

We evaluate the performance of Large Language Models (LLMs) on numerical reasoning tasks in Hindi using the HI-REASO benchmark. As illustrated in Figure 1, the evaluation pipeline begins with a Hindi numerical reasoning question, which is processed using one of three prompting strategies: Hindi Native Prompting (HNaP), Cross-Lingual Prompting (XLP; Qin et al., 2023), or Cross-Lingual Chain-of-Thought Prompting (XCoT; Huang et al., 2023). The prompted input is then provided to the LLM, whose generated response is post-processed through answer extraction and evaluated against gold answers.

5.1 Large Language Models

We evaluate Mathstral-7B (Mistral AI Team, 2024), Llama-3.3-70B (Meta AI, 2024), DeepSeek-R1-Distill-Llama-70B (DeepSeek-AI, 2025), Qwen2.5-Math-72B (Yang et al., 2024b), InternLM-Math-7B (Ying et al., 2024a), GPT-

4o (OpenAI, 2024), Gemini-2.0-Flash (Google Cloud, 2025), GPT-5.2 (OpenAI, 2025), Gemini-3.1-Pro (Google DeepMind, 2026), and Claude-Opus-4.5 (Anthropic, 2025). For consistency, we use instruction-tuned variants of all models. Across all experiments, we fix the temperature and top- p values to 1.0 and apply a standardized prompt template (see Appendix G.3) to ensure fair comparison across models and prompting strategies.

5.2 Prompting Techniques

Prompt design plays a crucial role in LLM performance, as models are highly sensitive to prompt formulation (Zhao et al., 2021). Given the strong zero-shot capabilities of modern LLMs (Wei et al., 2021; Kojima et al., 2022), we evaluate three zero-shot prompting strategies.

Cross-Lingual Chain-of-Thought Prompting (XCoT). XCoT extends chain-of-thought reasoning (Wei et al., 2022) to multilingual settings, enabling LLMs to perform step-by-step reasoning across languages (Huang et al., 2023; Shi et al., 2022b). Following prior work, the Hindi question is first translated into English, after which the model generates an explicit reasoning chain before producing the final answer in a standardized format.

Cross-Lingual Prompting (XLP). XLP enables LLMs to solve numerical reasoning problems across languages by translating the input while preserving its underlying mathematical structure (Qin et al., 2023). The model directly outputs the final answer without explicit intermediate reasoning steps.

Hindi Native Prompting (HNaP). Hindi Native Prompting evaluates LLMs’ ability to directly perform numerical reasoning in Hindi. Both instructions and questions are presented in Hindi, and models are required to generate answers in a fixed Hindi format. This setting provides a direct assessment of language-specific numerical reasoning without cross-lingual assistance. Detailed prompt templates are provided in Appendix G.

5.3 Evaluation Protocol

For robust evaluation, each model is tested using two independent runs, each on a randomly sampled subset of 920 instances from HI-REASO. The sampling ensures balanced task coverage, with

130 instances each from CA, CQ, FiB, QNLI, and AWP, and 270 instances from DS. Final results report the average accuracy across both runs (see Table 3), with per-run results provided in Appendix G. Model performance is measured using Exact-Match accuracy, following standard practice in numerical reasoning benchmarks (Dua et al., 2019; Qin et al., 2023).

Human Baseline. To establish a human performance reference, we randomly sampled 300 test instances with balanced coverage across all six task categories. Each instance was independently solved by three native Hindi speakers with strong mathematical backgrounds. Human accuracy is reported as the average across annotators, serving as an upper bound and highlighting the performance gap between LLMs and human numerical reasoning.

6 Results and Discussion

We evaluate LLM performance on HI-REASO to analyze model behavior across prompting strategies and task categories (see Table 3). The results reveal clear performance stratification across models, prompting methods, and reasoning types. To further disentangle the effects of model scale and training procedures, we complement our main analysis with a controlled scaling study within a single model family; detailed experimental setup, scaling trends, and distillation effects are presented in Appendix F. Also, a detailed error analysis examining failure modes across task categories is provided in Appendix D.1.

6.1 Performance Across Prompting Strategies

Large, general-purpose models such as GPT-4o and Gemini-2.0-Flash consistently achieve strong performance across all prompting strategies. Gemini-2.0-Flash attains average accuracies of 87.31, 89.63, and 83.02 under Hindi Native Prompting (HNaP), Cross-Lingual Prompting (XLP), and Cross-Lingual Chain-of-Thought Prompting (XCoT), respectively, with GPT-4o closely matching these results. These models demonstrate robustness to prompt variation, reflecting strong multilingual and numerical reasoning capabilities.

In contrast, smaller or distilled models exhibit substantially lower performance, particularly under HNaP. Mathstral-7B and DeepSeek-R1-

Distill-Llama-70B achieve accuracies of 17.59% and 14.78%, respectively, under this setting. However, both models benefit markedly from XLP, with Mathstral-7B improving to 42.32% and DeepSeek-R1-Distill-Llama-70B to 51.91%. This trend suggests that cross-lingual prompts provide linguistic and structural cues that better align with the models’ pretraining distributions.

Despite its larger size, DeepSeek-R1-Distill-Llama-70B underperforms Llama-3.3-70B under Hindi prompts and, in some cases, even trails Mathstral-7B. Qualitative inspection indicates difficulties in following Hindi instructions and generating coherent Hindi output. This behavior is consistent with prior findings that distillation can trade off multilingual generalization in favor of reasoning performance in high-resource languages (Payoungkhamdee et al., 2024; Diddee et al., 2022).

6.2 Task-wise Performance Trends

Domain-Specific (DS) and Quantitative Natural Language Inference (QNLI) remain the most challenging task categories across models. Earlier models such as GPT-4o and Gemini-2.0-Flash show relatively lower performance on DS and QNLI compared with arithmetic-oriented tasks, while newer frontier models improve results substantially. For example, Gemini-3.1-Pro reaches 90.20 on DS under HNaP and 79.00 on QNLI under XLP, and GPT-5.2 reaches 89.20 on DS under HNaP and 78.33 on QNLI under both XLP and XCoT. However, these scores still remain below the human baseline, particularly on QNLI, where human performance is 99.45. In contrast, models perform substantially better on Arithmetic Word Problems (AWP), Fill-in-the-Blanks (FiB), and Commonsense Arithmetic (CA), reflecting stronger capabilities in standard arithmetic computation and proportional reasoning. Under XLP, Gemini-3.1-Pro achieves 99.40 on AWP, 99.00 on CA, and 98.00 on FiB, while Gemini-2.0-Flash also performs strongly on these categories.

6.3 Impact of Cross-Lingual Prompting

Switching from HNaP to XLP yields consistent performance gains, particularly for lower- and mid-tier models, highlighting the effectiveness of cross-lingual cues in mitigating language-specific limitations. However, XCoT does not uniformly improve performance across models and tasks, suggesting that multilingual chain-of-thought reasoning requires further refinement to be effective in

Table 3: Comparison of LLM performance across prompting strategies on HI-REASO. Results report average Exact-Match accuracy (%) over two evaluation runs for six task categories. The best results are **orange boldfaced** and second-best results are **gray underlined**.

Models	CA	DS	CQ	FiB	QNLI	AWP	Avg.
<i>HNAP</i>							
Mathstral-7B	16.67	12.60	24.67	16.67	21.33	13.00	17.49
Llama-3.3-70B	67.67	34.00	60.33	50.33	27.67	51.33	48.56
DeepSeek-R1-Distill-Llama-70B	19.67	11.00	18.67	12.67	15.67	11.00	14.78
Qwen2.5-Math-72B	73.00	36.00	68.00	56.00	24.00	55.60	52.10
InternLM-Math-7B	60.00	27.00	55.00	41.00	18.00	46.80	41.30
GPT-4o	94.00	68.40	93.00	86.67	66.33	90.33	83.13
Gemini-2.0-Flash	98.00	84.20	91.67	90.00	64.00	<u>96.00</u>	87.31
GPT-5.2	<u>98.67</u>	<u>89.20</u>	<u>96.33</u>	<u>92.67</u>	<u>74.33</u>	94.80	<u>91.00</u>
Gemini-3.1-Pro	99.00	90.20	97.00	94.33	76.67	96.60	92.30
Claude-Opus-4.5	97.33	86.40	94.67	90.33	72.67	95.00	89.40
<i>XLP</i>							
Mathstral-7B	45.33	27.20	54.33	41.33	45.33	40.33	42.32
Llama-3.3-70B	92.00	71.60	94.00	86.67	63.00	92.00	83.21
DeepSeek-R1-Distill-Llama-70B	75.33	37.80	45.67	59.67	36.00	57.00	51.91
Qwen2.5-Math-72B	94.00	63.00	92.00	84.00	51.00	85.20	78.20
InternLM-Math-7B	82.00	50.00	78.00	70.00	35.00	67.80	63.80
GPT-4o	95.33	66.00	96.00	83.67	71.67	86.67	83.56
Gemini-2.0-Flash	97.00	78.80	96.67	96.00	67.33	<u>99.00</u>	89.63
GPT-5.2	<u>98.67</u>	<u>84.60</u>	<u>98.33</u>	<u>97.67</u>	<u>78.33</u>	96.80	<u>92.40</u>
Gemini-3.1-Pro	99.00	87.20	99.00	98.00	79.00	99.40	93.60
Claude-Opus-4.5	97.00	83.60	97.00	95.33	74.67	99.00	91.10
<i>XCoT</i>							
Mathstral-7B	46.00	28.20	57.67	38.00	43.67	44.00	42.93
Llama-3.3-70B	92.33	66.60	90.67	83.67	59.00	88.67	80.16
DeepSeek-R1-Distill-Llama-70B	67.67	28.00	58.00	45.00	31.67	42.33	45.45
Qwen2.5-Math-72B	92.33	58.60	91.67	75.33	49.33	86.00	75.54
InternLM-Math-7B	79.67	44.20	74.33	61.33	33.67	62.33	59.26
GPT-4o	96.33	61.60	95.33	75.67	69.33	83.67	80.32
Gemini-2.0-Flash	93.33	72.00	98.67	85.33	65.00	85.00	83.02
GPT-5.2	<u>96.67</u>	<u>82.20</u>	97.00	<u>91.00</u>	<u>78.33</u>	<u>98.40</u>	<u>90.60</u>
Gemini-3.1-Pro	97.33	83.20	<u>97.67</u>	91.33	78.67	99.00	91.20
Claude-Opus-4.5	95.00	79.60	95.33	88.00	76.67	99.40	89.00
<i>Human Baseline (Avg., N=3)</i>							
	99.61	97.74	100.00	97.72	99.45	99.89	99.40

low-resource language settings.

6.4 Gap to Human Performance

Despite strong model performance, a clear gap remains compared to human annotators, who achieve near-perfect accuracy (99.40% average). Models struggle most on Domain-Specific (DS) and Quantitative Natural Language Inference (QNLI) tasks. We report the *Language Robustness Index* (LRI) and *Reasoning Residual Gap* (RRG) in Appendix E to further analyze these limitations.

7 Related Work

Numerical Reasoning Benchmarks. Numerical reasoning in natural language has been studied through a wide range of benchmarks that vary in scale, complexity, and task formulation. Early datasets focused on mapping text to arithmetic or algebraic expressions, enabling models to solve structured word problems (Kushman et al., 2014; Hosseini et al., 2014; Koncel-Kedziorski et al., 2015). These benchmarks established core formulations for numerical reasoning but were limited in scale and linguistic diversity.

Subsequent work expanded both dataset size and reasoning complexity. Larger benchmarks enabled neural models to address compositional and multi-step reasoning (Ling et al., 2017; Dua et al., 2019), while datasets such as MathQA introduced diverse problem structures and operation-based supervision (Amini et al., 2019). Other efforts examined numerical representation challenges, including scale sensitivity in embeddings (Zhang et al., 2020), and increased problem difficulty through competition-level datasets such as GSM8K and MATH (Cobbe et al., 2021; Hendrycks et al., 2021). More recently, unified benchmarks such as NumGLUE and LILA have provided systematic evaluations across multiple numerical reasoning tasks (Mishra et al., 2022b,a). Although these benchmarks are comprehensive, they are largely English-centric. Multilingual extensions, including MGSM, demonstrate that numerical reasoning performance varies substantially across languages, even when models are evaluated on parallel problems (Shi et al., 2022a). However, existing multilingual benchmarks offer limited diagnostic insight into language-specific numerical reasoning behavior. To ensure that HI-REASO is not a derivative or translated version of existing benchmarks, we perform a comprehensive overlap analysis with prior datasets; details of the methodology and findings are provided in Appendix C

Models and Multilingual Numerical Reasoning.

Alongside benchmark development, significant effort has focused on improving numerical reasoning capabilities in large language models (Ahn et al., 2024). General-purpose LLMs such as GPT-3 and GPT-4 achieve strong language understanding performance but exhibit consistent weaknesses on complex numerical reasoning tasks (Brown et al., 2020; Achiam et al., 2023). Prompting methods, particularly chain-of-thought prompting, have been shown to improve numerical reasoning by eliciting intermediate steps (Wei et al., 2022). Nevertheless, these gains are limited and degrade in multilingual settings, where reasoning accuracy varies across languages (Shi et al., 2022a; Qin et al., 2023; Huang et al., 2023). Recent work has introduced math-specialized LLMs aimed at improving numerical reasoning accuracy, including Llemma, WizardMath, DeepSeekMath, Qwen2.5-Math, and InternLM-Math (Azerbayev et al., 2023; Luo et al., 2023; Shao et al., 2024; Yang et al., 2024a; Ying et al., 2024b). While these models

perform well on English benchmarks, their behavior under non-English and low-resource language conditions remains insufficiently studied.

Indic and Hindi Evaluation. Despite Hindi being one of the most widely spoken languages globally, evaluation resources for numerical reasoning in Hindi are limited. Existing Indic benchmarks such as IndicGLUE and IndicXNLI support general natural language understanding evaluation across Indic languages (Sankalp et al., 2025; Aggarwal et al., 2022; Khanuja et al., 2020), but they do not explicitly evaluate numerical reasoning. Consequently, the extent to which LLMs can perform numerical reasoning under Hindi-native conditions remains unclear.

HI-REASO addresses this gap by introducing a benchmark focused specifically on numerical reasoning in Hindi. By evaluating models under Hindi-native and cross-lingual prompting settings, our benchmark enables a controlled analysis of language-dependent numerical reasoning failures and complements existing English-centric and multilingual evaluations.

8 Conclusion

We introduced HI-REASO, a benchmark comprising six diverse task categories designed to evaluate the numerical reasoning capabilities of Large Language Models (LLMs) in Hindi, a low-resource language with limited computational linguistic support. Through extensive evaluation, we observed a substantial gap between state-of-the-art LLMs and human performance across multiple numerical reasoning tasks, particularly in domain-specific reasoning and quantitative inference. By providing a structured and challenging evaluation framework, HI-REASO contributes toward advancing equitable numerical reasoning research for underrepresented languages.

Limitations. Despite its contributions, HI-REASO has several limitations. First, while the dataset covers a range of task types, its scale can be expanded to include a broader variety of problem formulations and real-world contexts. Second, the current focus on Hindi limits cross-lingual analysis; extending the benchmark to additional low-resource languages would enable more comprehensive multilingual evaluation. Third, a deeper comparison with existing numerical reasoning benchmarks and a larger human evaluation

set would provide stronger contextualization of model performance. Finally, our experiments focus on zero-shot prompting strategies; future work could explore few-shot settings, more optimized chain-of-thought prompting, and newer reasoning-oriented LLMs to further enhance evaluation depth.

Ethical Considerations. The HI-REASO dataset was curated by native Hindi-speaking researchers and trained student annotators aged 22–26, each with prior research experience. Detailed annotation guidelines were provided to ensure consistency, covering numerical reasoning principles, data sources, task formatting, and verification procedures. All annotations underwent cross-checking and manual validation, and annotators were compensated fairly in accordance with institutional standards. The dataset contains no personal or sensitive information and is intended solely for research purposes.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Divyanshu Aggarwal, Vivek Gupta, and Anoop Kunchukuttan. 2022. Indicxnli: Evaluating multilingual inference for indian languages. *arXiv preprint arXiv:2204.08776*.
- Janice Ahn, Rishu Verma, Renze Lou, Di Liu, Rui Zhang, and Wenpeng Yin. 2024. Large language models for mathematical reasoning: Progresses and challenges. *arXiv preprint arXiv:2402.00157*.
- Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: Human language technologies, volume 1 (long and short papers)*, pages 2357–2367.
- Anthropic. 2025. **Claude Opus 4.5 System Card**. System card. Official system card for Claude Opus 4.5.
- Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen McAleer, Albert Q Jiang, Jia Deng, Stella Biderman, and Sean Welleck. 2023. Llemma: An open language model for mathematics. *arXiv preprint arXiv:2310.10631*.
- Yejin Bang. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. **Language models are few-shot learners**. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- DeepSeek-AI. 2025. **DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning**. Preprint, arXiv:2501.12948.
- Harshita Diddee, Sandipan Dandapat, Monojit Choudhury, Tanuja Ganu, and Kalika Bali. 2022. Too brittle to touch: Comparing the stability of quantization and distillation towards developing low-resource mt models. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 870–885.
- Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. *arXiv preprint arXiv:1903.00161*.
- Google Cloud. 2025. **Gemini 2.0 Flash**. Google Cloud model documentation. Model ID: gemini-2.0-flash; release date February 5, 2025.
- Google DeepMind. 2026. **Gemini 3.1 Pro Model Card**. Model card. Published February 19, 2026.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.
- Mohammad Javad Hosseini, Hannaneh Hajishirzi, Oren Etzioni, and Nate Kushman. 2014. Learning to solve arithmetic word problems with verb categorization. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 523–533.
- 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*.

- Simran Khanuja, Sandipan Dandapat, Anirudh Srinivasan, Sunayana Sitaram, and Monojit Choudhury. 2020. Gluecos: An evaluation benchmark for code-switched nlp. *arXiv preprint arXiv:2004.12376*.
- 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.
- Rik Koncel-Kedziorski, Hannaneh Hajishirzi, Ashish Sabharwal, Oren Etzioni, and Siena Dumas Ang. 2015. Parsing algebraic word problems into equations. *Transactions of the Association for Computational Linguistics*, 3:585–597.
- Nate Kushman, Yoav Artzi, Luke Zettlemoyer, and Regina Barzilay. 2014. Learning to automatically solve algebra word problems. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 271–281.
- Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. *arXiv preprint arXiv:1705.04146*.
- Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. 2023. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *arXiv preprint arXiv:2308.09583*.
- Meta AI. 2024. [Llama 3.3 70B Instruct Model Card](#). Model card. Llama 3.3 70B Instruct, released December 6, 2024.
- Swaroop Mishra, Matthew Finlayson, Pan Lu, Leonard Tang, Sean Welleck, Chitta Baral, Tanmay Rajpurohit, Oyvind Tafford, Ashish Sabharwal, Peter Clark, and 1 others. 2022a. Lila: A unified benchmark for mathematical reasoning. *arXiv preprint arXiv:2210.17517*.
- Swaroop Mishra, Arindam Mitra, Neeraj Varshney, Bhavdeep Sachdeva, Peter Clark, Chitta Baral, and Ashwin Kalyan. 2022b. Numglue: A suite of fundamental yet challenging mathematical reasoning tasks. *arXiv preprint arXiv:2204.05660*.
- Mistral AI Team. 2024. [Mathstral](#). Mistral AI Research Release. Official release page for Mathstral 7B.
- OpenAI. 2024. [GPT-4o System Card](#). Preprint, arXiv:2410.21276.
- OpenAI. 2025. [Update to GPT-5 System Card: GPT-5.2](#). System card update. Official system card update for GPT-5.2 Instant and GPT-5.2 Thinking.
- Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? *arXiv preprint arXiv:2103.07191*.
- Patomporn Payoungkhamdee, Peerat Limkonchotiwat, Jinheon Baek, Potsawee Manakul, Can Udomcharoenchaikit, Ekapol Chuangsuwanich, and Sarana Nutanong. 2024. An empirical study of multilingual reasoning distillation for question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7739–7751.
- Libo Qin, Qiguang Chen, Fuxuan Wei, Shijue Huang, and Wanxiang Che. 2023. Cross-lingual prompting: Improving zero-shot chain-of-thought reasoning across languages. *arXiv preprint arXiv:2310.14799*.
- KJ Sankalp, Ashutosh Kumar, Laxmaan Balaji, Nikunj Kotecha, Vinija Jain, Aman Chadha, and Sreyoshi Bhaduri. 2025. Indicmmlu-pro: Benchmarking indic large language models on multi-task language understanding. *CoRR*.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.
- Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, and 1 others. 2022a. Language models are multilingual chain-of-thought reasoners. *arXiv preprint arXiv:2210.03057*.
- Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, and 1 others. 2022b. Language models are multilingual chain-of-thought reasoners. *arXiv preprint arXiv:2210.03057*.
- Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.
- An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, and 1 others. 2024a. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*.
- An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, and 1 others. 2024b. [Qwen2.5-Math Technical Report: Toward Mathematical Expert Model via Self-Improvement](#). Preprint, arXiv:2409.12122.

Huaiyuan Ying, Shuo Zhang, Linyang Li, Zhejian Zhou, Yunfan Shao, Zhaoye Fei, Yichuan Ma, Jiawei Hong, Kuikun Liu, Ziyi Wang, and 1 others. 2024a. [InternLM-Math: Open Math Large Language Models Toward Verifiable Reasoning](#). *Preprint*, arXiv:2402.06332.

Huaiyuan Ying, Shuo Zhang, Linyang Li, Zhejian Zhou, Yunfan Shao, Zhaoye Fei, Yichuan Ma, Jiawei Hong, Kuikun Liu, Ziyi Wang, and 1 others. 2024b. Internlm-math: Open math large language models toward verifiable reasoning. *arXiv preprint arXiv:2402.06332*.

Xikun Zhang, Deepak Ramachandran, Ian Tenney, Yanai Elazar, and Dan Roth. 2020. Do language embeddings capture scales? *arXiv preprint arXiv:2010.05345*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Zican Dong, Yupeng Hou, Beichen Zhang, Yingqian Min, Junjie Zhang, Peiyu Liu, Xiaolei Wang, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, and 4 others. 2026. [A survey of large language models](#). *Frontiers of Computer Science*, 20(12):2012627.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pages 12697–12706. PMLR.

A Difficulty Characterization

To better characterize the reasoning demands of HI-REASO, we annotated each instance with a three-level difficulty label: Easy, Medium, or Hard. Difficulty was assigned using a rule-based rubric that considers the number of reasoning steps, presence of implicit quantities, need for formula selection, unit conversion, domain-specific knowledge, and whether the answer requires mapping a computed value to a logical relation. An instance is labeled Easy if it requires a single arithmetic operation, direct numerical comparison, or retrieval of a common numerical fact. Medium problems require two-step arithmetic, proportional reasoning, basic unit conversion, or use of an explicitly indicated formula. Hard problems require multi-step reasoning, implicit quantity extraction, formula selection without direct cueing, domain-specific constants, or numerical inference over premise-hypothesis pairs.

This distribution Figure 2 shows that HI-REASO is not dominated by trivial arithmetic. While approximately one-third of the dataset consists of single-step or direct reasoning problems,

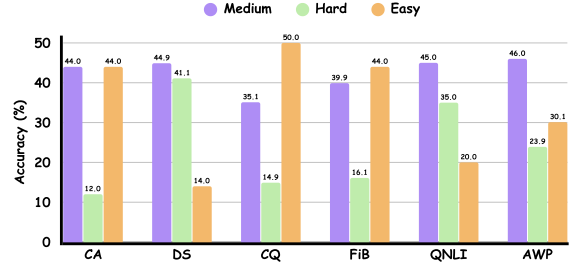


Figure 2: Difficulty distribution of HI-REASO.

nearly two-thirds require medium or hard reasoning. Domain-Specific and QNLI tasks are intentionally skewed toward harder examples, reflecting their reliance on formula selection, unit conversion, and logical inference

B Diversity Analysis

We also categorize each problem by its primary reasoning phenomenon. Unlike task labels, which describe the output format, this analysis captures the underlying numerical reasoning skill required to solve the problem.

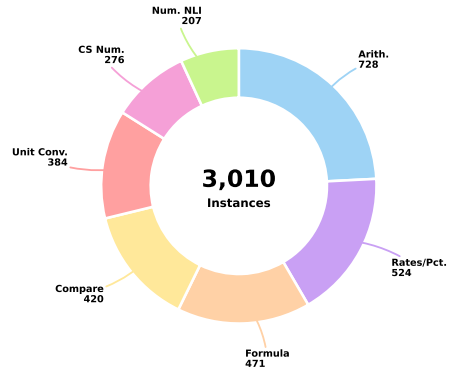


Figure 3: Primary reasoning-skill distribution.

The analysis in Figure 3 shows that HI-REASO covers a broad set of numerical reasoning phenomena rather than only arithmetic word problems. In particular, 28.4% of the dataset involves either unit conversion, measurement, or formula-based reasoning, while 20.9% involves comparison, ordering, or numerical entailment. These categories are underrepresented in many existing math word-problem benchmarks.

We further analyze linguistic and contextual diversity. HI-REASO includes problems expressed using school-style Hindi, everyday conversational contexts, formal STEM terminology, rupee-based commerce, measurement units, time and calendar quantities, and culturally grounded Hindi com-

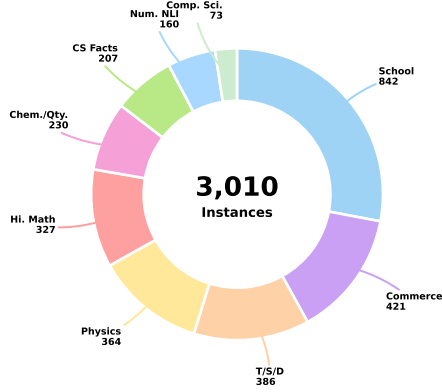


Figure 4: Context-domain distribution.

nonsense facts such as Devanagari vowels, Indian currency, and classroom objects described in Figure 4.

C Overlap with Existing Benchmarks

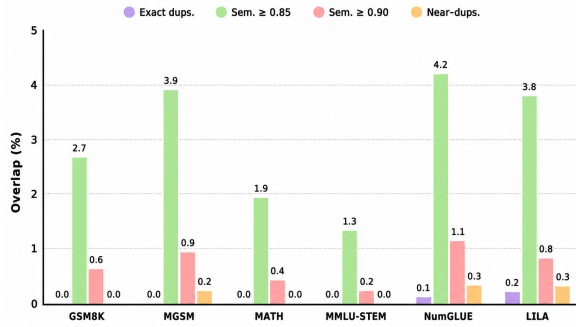


Figure 5: Overlap between HI-REASO and existing benchmarks.

To determine whether HI-REASO duplicates existing math benchmarks, we conduct both lexical and semantic overlap analysis against GSM8K, MGSM, MATH, MMLU-STEM, NumGLUE, and LILA. All problems are normalized by lowercasing, removing punctuation, normalizing numerals, removing unit markers where appropriate, and translating Hindi problems into English for semantic comparison. We then compute exact-match overlap and sentence-embedding similarity using multilingual SBERT. Candidate overlaps are manually inspected when cosine similarity exceeds 0.85. The maximum semantic overlap is 4.2%, and exact duplication is effectively absent. Manual inspection shows that most high-similarity cases share only a broad mathematical template, such as unit price calculation or percentage profit, but differ in language, cultural setting, quantities, and task formulation. Cases that were judged to be

genuine near-duplicates were removed or rewritten. These findings, summarized in Figure 5, substantiate the claim that HI-REASO is not merely a Hindi translation of existing English benchmarks. Its novelty comes from three sources: Hindi-native problem formulation, culturally grounded numerical contexts, and the inclusion of task types such as QNLI, CQ, and FiB that are not jointly covered by prior math reasoning datasets.

C.1 Difficulty Validates Model Behavior

We also analyze whether the proposed difficulty labels correlate with model performance. As reported in Figure 6, accuracy decreases monotonically from Easy to Hard across all model groups, while humans remain near ceiling throughout.

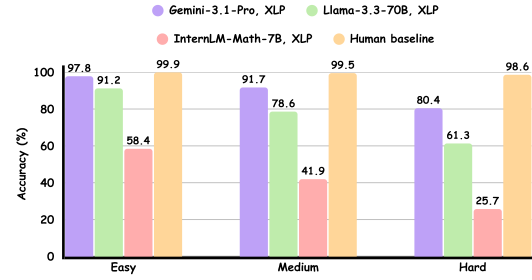


Figure 6: Accuracy trends of models and humans across increasing task difficulty.

This analysis demonstrates that the proposed difficulty labels are meaningful: accuracy decreases monotonically from Easy to Hard for all model groups, while humans remain near ceiling.

D Error Analysis

D.1 Quantitative Error Analysis

To better understand the limitations of different prompting strategies and model behaviors, we conduct a detailed error analysis, summarized in Table 4 & Table 5. We categorize errors into two types: *wrong format* (wf%), which captures outputs that deviate from the expected answer format, and *wrong calculation* (wc%), which captures arithmetic or logical errors among correctly formatted outputs. While wf% reflects failures in instruction adherence, wc% isolates reasoning errors independent of formatting issues.

Table 4: Breakdown of error rates (Run 1) across HNaP, XLP, and XCoT prompting strategies. $wf\%$ denotes wrong-format errors and $wc\%$ denotes wrong-calculation errors; lower values indicate better performance. Avg. represents the mean error across all tasks. The best results are **orange boldfaced** and second-best results are gray underlined

Models	Error Type	CA	DS	CQ	FiB	QNLI	AWP	Avg.
<i>HNaP</i>								
Mathstral-7B	wf%	46.00	56.40	65.33	53.33	52.00	56.67	54.96
	wc%	71.82	77.73	30.54	73.64	56.67	80.00	65.07
Llama-3.3-70B	wf%	22.67	58.40	30.67	40.00	63.33	42.00	42.84
	wc%	8.81	19.11	6.85	12.67	12.50	1.11	10.17
DeepSeek-R1-Distill-Llama-70B	wf%	75.73	80.31	73.27	79.91	78.63	80.71	78.09
	wc%	6.12	23.33	12.22	15.00	43.85	3.45	17.33
Qwen2.5-Math-72B	wf%	10.20	28.40	16.00	20.00	22.10	13.10	18.30
	wc%	6.10	22.70	7.40	13.90	28.20	4.00	13.72
InternLM-Math-7B	wf%	24.50	39.20	28.10	35.70	38.40	24.90	31.80
	wc%	19.00	45.20	24.30	43.10	56.40	50.20	39.70
GPT-4o	wf%	2.85	0.40	5.66	4.11	3.02	6.44	3.75
	wc%	3.70	23.04	4.38	11.83	31.22	2.68	12.81
Gemini-2.0-Flash	wf%	2.23	1.50	0.00	1.13	0.00	0.00	0.81
	wc%	2.67	12.45	5.33	10.95	35.33	1.33	11.34
GPT-5.2	wf%	<u>1.85</u>	<u>0.30</u>	1.40	2.20	1.80	1.60	1.53
	wc%	<u>1.80</u>	<u>10.90</u>	<u>3.20</u>	8.90	25.40	1.20	<u>8.57</u>
Gemini-3.1-Pro	wf%	1.40	0.20	<u>1.10</u>	<u>1.70</u>	<u>1.20</u>	<u>1.20</u>	<u>1.13</u>
	wc%	1.50	9.70	2.80	8.20	<u>23.90</u>	0.90	7.83
Claude-Opus-4.5	wf%	2.05	0.60	1.90	2.60	2.10	2.00	1.88
	wc%	2.10	11.60	3.60	9.40	26.10	<u>1.10</u>	8.98
<i>XLP</i>								
Mathstral-7B	wf%	22.00	24.80	16.67	20.00	7.33	24.67	19.25
	wc%	45.88	61.17	31.82	42.38	45.65	40.00	44.48
Llama-3.3-70B	wf%	4.92	3.08	6.55	2.49	1.33	5.77	4.02
	wc%	5.65	20.97	3.10	9.44	29.85	5.07	12.35
DeepSeek-R1-Distill-Llama-70B	wf%	14.67	46.00	42.00	28.67	46.67	24.00	33.67
	wc%	5.04	27.27	10.83	12.83	25.38	9.19	15.09
Qwen2.5-Math-72B	wf%	6.10	8.20	5.30	6.80	10.40	4.10	6.82
	wc%	3.80	22.40	4.90	10.70	28.60	5.80	12.70
InternLM-Math-7B	wf%	12.40	18.90	13.00	17.60	18.50	8.40	14.80
	wc%	10.30	31.80	12.50	25.40	45.80	25.40	25.20
GPT-4o	wf%	6.33	1.20	2.94	4.27	3.69	5.11	3.92
	wc%	1.03	27.39	4.09	13.08	26.49	6.20	13.05
Gemini-2.0-Flash	wf%	3.22	6.48	2.61	4.95	5.38	2.73	4.23
	wc%	5.06	21.98	1.33	4.09	33.24	0.74	11.07
GPT-5.2	wf%	<u>2.05</u>	<u>0.90</u>	<u>1.80</u>	<u>2.30</u>	2.70	<u>1.80</u>	<u>1.93</u>
	wc%	<u>0.90</u>	<u>17.80</u>	2.40	6.20	<u>23.90</u>	1.30	<u>8.75</u>
Gemini-3.1-Pro	wf%	1.70	0.70	1.40	2.00	<u>2.10</u>	1.50	1.57
	wc%	0.70	16.90	<u>2.00</u>	<u>5.50</u>	22.70	<u>1.00</u>	8.13
Claude-Opus-4.5	wf%	2.40	1.10	2.10	2.60	2.90	2.00	2.18
	wc%	1.10	18.40	2.80	6.80	24.80	1.50	9.23
<i>XCoT</i>								
Mathstral-7B	wf%	18.67	27.60	9.33	28.67	7.33	20.00	18.60
	wc%	38.60	57.95	31.32	43.26	44.84	34.76	41.79
Llama-3.3-70B	wf%	2.58	6.17	0.67	3.41	14.00	4.88	5.28
	wc%	3.19	25.09	1.19	12.86	27.72	4.18	12.37
DeepSeek-R1-Distill-Llama-70B	wf%	22.00	61.20	42.00	44.00	55.33	46.67	45.20
	wc%	0.78	24.72	3.89	10.29	20.77	11.54	12.00
Qwen2.5-Math-72B	wf%	6.10	9.40	5.20	7.80	10.30	4.40	7.20
	wc%	3.60	24.90	4.20	11.80	27.70	4.60	12.80
InternLM-Math-7B	wf%	13.20	20.40	12.50	17.60	19.00	8.50	15.20
	wc%	10.10	32.50	11.40	23.80	45.10	29.50	25.40
GPT-4o	wf%	5.92	12.80	2.36	6.67	6.74	0.67	5.86
	wc%	0.42	23.68	3.01	10.00	27.86	6.42	11.90
Gemini-2.0-Flash	wf%	3.07	3.60	4.62	2.67	1.33	6.67	3.66
	wc%	6.29	17.78	0.74	0.69	27.59	2.00	9.18
GPT-5.2	wf%	<u>2.40</u>	<u>2.80</u>	1.70	<u>2.10</u>	2.40	1.30	<u>2.12</u>
	wc%	<u>0.40</u>	<u>16.20</u>	1.10	6.00	22.40	<u>1.20</u>	<u>7.88</u>
Gemini-3.1-Pro	wf%	2.00	2.30	<u>1.40</u>	1.80	<u>2.00</u>	<u>1.10</u>	1.77
	wc%	0.30	15.30	<u>0.90</u>	<u>5.40</u>	<u>21.60</u>	0.90	7.40
Claude-Opus-4.5	wf%	2.70	3.10	2.00	2.50	2.80	1.60	2.45
	wc%	0.50	16.90	1.30	6.50	23.20	1.40	8.30

Formally, we define the following quantities:

$$WP = T - EM \quad (1)$$

$$WF\% = \frac{WF}{T} \times 100 \quad (2)$$

$$CF = T - WF \quad (3)$$

$$WC\% = \frac{WP - WF}{CF} \times 100 \quad (4)$$

where T denotes the total number of samples, EM denotes exact matches, WP denotes wrong predictions, WF denotes wrong-format outputs, CF denotes correctly formatted outputs, $WF\%$ denotes the percentage of formatting errors, and $WC\%$ denotes the percentage of reasoning errors among correctly formatted predictions.

To ensure robustness, we perform two independent evaluation runs for each experimental setting. Error statistics for each model and prompting strategy are reported in Table 4 & Table 5.

D.2 Qualitative Error Analysis

To complement the quantitative results, we provide a qualitative, operation-level error analysis grounded in HI-REASO task definitions and summarized in Table 6 and Figure 7. We observe that model failures consistently decompose into reasoning errors ($wc\%$) and instruction-following errors ($wf\%$), with their relative contribution varying by task. In simpler tasks such as common-sense arithmetic (CA) and comparison (CQ), reasoning is often correct, but performance is disproportionately affected by format non-adherence under Hindi Native Prompting (e.g., omission of the required answer template or option identifier), indicating $wf\%$ as the dominant bottleneck. In contrast, tasks involving formula application and unit conversion (DS) exhibit primarily wc -type failures, including incorrect formula selection, misuse of constants (e.g., π), rounding inaccuracies, and unit inconsistencies—consistent with their higher error rates and status as the most challenging category. More complex settings such as QNLI and arithmetic word problems (AWP) reveal dual failure modes, where numerical reasoning errors (e.g., division, ratio setup, or incorrect percentage base) interact with difficulties in mapping computed values to logical relations or required output units. The operation-level breakdown in Figure 7 further shows that multi-step chaining (47.9%) and unit conversion (42.5%) are particularly error-prone due to error propagation and ambiguity in Hindi

quantifiers, while proportional reasoning (39.1%) and formula-based tasks (33.7%) suffer from structural setup errors and inconsistent constant usage. Overall, this analysis demonstrates that performance limitations arise not only from arithmetic weaknesses but from the interaction between compositional reasoning complexity and strict output constraints, with $wf\%$ dominating in structurally simple tasks and $wc\%$ driving errors in conceptually demanding ones.

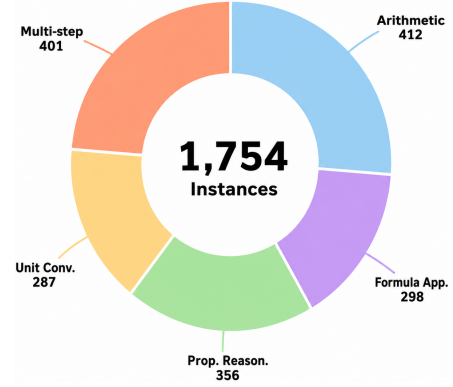


Figure 7: Error analysis of GPT-4o across mathematical tasks using HNaP.

E Language and Reasoning Bottleneck Indices

A central objective of HI-REASO is to distinguish between failures arising from language realization and those arising from limitations in numerical reasoning. The evaluation protocol incorporates multiple prompting conditions designed to isolate these effects. In this section, we formalize two aggregate diagnostic indices that quantify language robustness and residual reasoning limitations across models.

Across models, non-zero RRG values indicate that reasoning limitations persist even after

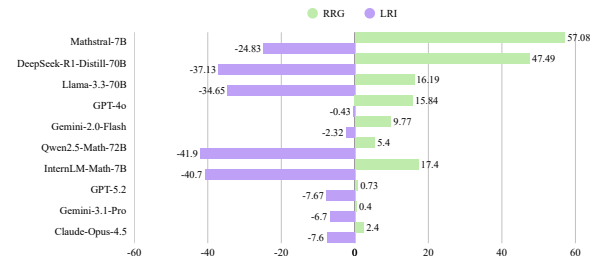


Figure 8: Diagnostic analysis of reasoning and language sensitivity across models using RRG (bigger; closer to zero value is better) and LRI (smaller value is better).

Table 5: Error analysis breakdown by prompting strategy (Run 2). We report wrong-format (*wf%*) and wrong-calculation (*wc%*) error rates; lower values indicate better format compliance and reasoning accuracy. Avg. denotes the mean error across all tasks. The best results are **orange boldfaced** and second-best results are **gray underlined**

Models	Error Type	CA	DS	CQ	FiB	QNLI	AWP	Avg.
<i>HNaP</i>								
Mathstral-7B	wf%	32.33	41.80	54.33	37.00	40.33	49.00	42.47
	wc%	70.04	73.86	26.00	64.33	56.41	72.67	60.55
Llama-3.3-70B	wf%	11.00	48.20	19.67	29.67	59.67	38.33	34.42
	wc%	3.92	2.95	2.08	2.83	12.25	2.67	4.45
DeepSeek-R1-Distill-Llama-70B	wf%	69.67	79.80	70.33	77.67	71.67	80.33	74.91
	wc%	0.00	0.00	17.00	0.00	16.09	78.63	18.62
Qwen2.5-Math-72B	wf%	9.80	26.70	15.10	18.20	20.60	12.30	17.12
	wc%	5.60	18.20	6.40	9.50	25.90	4.20	11.63
InternLM-Math-7B	wf%	22.10	36.80	26.30	33.10	36.90	23.70	29.82
	wc%	18.20	43.10	23.60	41.00	54.90	50.10	38.48
GPT-4o	wf%	2.26	3.27	2.09	2.66	3.52	2.08	2.65
	wc%	2.60	16.75	3.95	3.63	26.33	2.66	9.32
Gemini-2.0-Flash	wf%	3.77	0.81	0.00	1.76	0.00	2.91	1.54
	wc%	2.47	7.72	2.33	4.87	27.67	3.01	8.01
GPT-5.2	wf%	<u>1.70</u>	<u>0.60</u>	<u>0.90</u>	<u>1.10</u>	<u>1.40</u>	<u>1.20</u>	<u>1.15</u>
	wc%	1.20	2.40	<u>1.90</u>	1.80	14.60	<u>2.10</u>	<u>4.00</u>
Gemini-3.1-Pro	wf%	1.30	0.40	0.70	0.90	1.10	0.90	0.88
	wc%	<u>0.90</u>	<u>2.20</u>	1.70	<u>1.50</u>	<u>13.40</u>	1.40	3.52
Claude-Opus-4.5	wf%	2.10	0.90	1.30	1.50	1.60	1.50	1.48
	wc%	1.50	3.10	2.10	2.30	15.50	2.40	4.48
<i>XLP</i>								
Mathstral-7B	wf%	17.67	13.80	6.33	12.33	3.00	18.33	11.91
	wc%	16.79	57.79	19.39	38.28	37.41	36.32	34.33
Llama-3.3-70B	wf%	2.28	2.29	4.54	3.81	4.42	4.19	3.59
	wc%	3.61	13.44	4.92	1.83	24.57	3.14	8.59
DeepSeek-R1-Distill-Llama-70B	wf%	3.77	8.41	0.00	8.33	0.00	7.34	4.64
	wc%	3.66	14.33	4.49	3.86	23.86	4.59	9.13
Qwen2.5-Math-72B	wf%	5.80	7.90	4.80	6.20	9.80	3.90	6.40
	wc%	3.40	16.80	4.20	8.90	25.30	4.80	10.57
InternLM-Math-7B	wf%	11.80	17.90	12.20	16.50	18.10	8.20	14.12
	wc%	9.80	30.20	11.90	23.90	43.60	24.30	23.95
GPT-4o	wf%	3.73	4.11	2.14	2.68	2.87	2.24	2.96
	wc%	2.70	16.91	2.30	2.83	20.33	0.57	7.61
Gemini-2.0-Flash	wf%	3.91	3.09	3.11	2.63	2.80	4.81	3.39
	wc%	3.94	9.46	3.83	2.51	21.94	4.19	7.65
GPT-5.2	wf%	<u>1.90</u>	<u>1.80</u>	<u>1.70</u>	<u>1.90</u>	<u>2.00</u>	<u>1.70</u>	<u>1.83</u>
	wc%	<u>2.10</u>	<u>8.20</u>	<u>2.20</u>	<u>1.70</u>	<u>19.80</u>	1.10	5.85
Gemini-3.1-Pro	wf%	1.50	1.40	<u>1.30</u>	1.60	<u>1.70</u>	1.30	1.47
	wc%	1.80	7.60	1.90	1.40	18.90	<u>0.90</u>	5.42
Claude-Opus-4.5	wf%	2.20	2.10	2.00	2.20	2.30	2.00	2.13
	wc%	2.40	8.80	2.60	2.10	20.40	1.30	6.27
<i>XCoT</i>								
Mathstral-7B	wf%	11.67	14.20	7.00	13.67	2.49	17.00	11.01
	wc%	30.04	56.45	16.14	40.41	45.00	35.17	37.20
Llama-3.3-70B	wf%	3.14	4.97	3.92	3.67	2.33	4.05	3.68
	wc%	4.53	20.73	4.33	0.29	16.59	2.69	8.19
DeepSeek-R1-Distill-Llama-70B	wf%	14.33	56.20	15.67	39.67	46.33	47.00	36.53
	wc%	2.10	11.65	2.95	0.35	25.23	2.80	7.51
Qwen2.5-Math-72B	wf%	5.60	8.70	5.10	7.10	9.70	4.20	6.73
	wc%	3.70	18.60	4.00	8.20	25.60	4.20	10.72
InternLM-Math-7B	wf%	12.70	19.60	11.90	16.90	18.50	8.30	14.65
	wc%	9.60	29.80	11.20	22.70	42.90	28.50	24.12
GPT-4o	wf%	2.63	2.60	4.83	0.33	4.63	2.94	2.99
	wc%	3.97	14.67	3.19	0.83	16.00	4.74	7.23
Gemini-2.0-Flash	wf%	3.38	2.79	2.74	0.33	3.68	2.79	2.62
	wc%	3.75	6.24	4.69	3.20	22.84	2.66	7.23
GPT-5.2	wf%	<u>2.10</u>	<u>2.20</u>	<u>2.00</u>	<u>0.27</u>	<u>2.10</u>	<u>1.90</u>	<u>1.76</u>
	wc%	<u>2.00</u>	<u>6.10</u>	<u>2.80</u>	<u>0.27</u>	<u>15.40</u>	<u>2.20</u>	<u>4.80</u>
Gemini-3.1-Pro	wf%	1.70	1.90	1.60	0.25	1.80	1.60	1.48
	wc%	1.80	5.70	2.50	0.25	14.80	2.00	4.51
Claude-Opus-4.5	wf%	2.40	2.50	2.30	0.31	2.60	2.20	2.05
	wc%	2.30	6.50	3.00	0.55	16.10	2.50	5.16

language complexity is reduced. This observation supports the interpretation that performance deficits arise from both language sensitivity and underlying numerical reasoning challenges.

Table 6: Qualitative failure analysis by task category, grounded in HI-REASO task definitions and released examples. wc% denotes reasoning errors among correctly formatted outputs; wf% denotes instruction non-adherence.

Task	Example (Hindi → Answer)	wc% Failure Modes	wf% Failure Modes	Key Insight
CA	एक सप्ताह में 7 दिन... 3 सप्ताह में? → Ans: 21	Incorrect multiplication (3×7); misreading unit relationship (days vs. weeks)	Correct numeral produced but missing required उत्तर: [संख्या] wrapper under HNaP	Arithmetic is straightforward; format non-adherence is the primary bottleneck for weaker models
DS	12 रेडियन को डिग्री में... → Ans: 687.55° त्रिज्या 5 मीटर... क्षेत्रफल? → Ans: 78.54 m ²	Wrong formula (degrees conversion or πr^2); imprecise constant (π); rounding errors; incorrect unit conversions	Less common for strong models; more frequent for smaller models under HNaP due to domain-specific output constraints	Hardest task across all models including GPT-4o and Gemini; compounding reasoning errors explain persistently high wc% even when wf% is low
CQ	12345 क्या 54321 से छोटी है या बड़ी? (1) छोटी (2) बड़ी → Ans: (1)	Reversed comparison direction; place-value confusion with large numbers	Outputs छोटी instead of required option-ID format उत्तर: (1), masking an otherwise correct judgment	Comparison logic is often sound; errors are disproportionately driven by format non-compliance under HNaP
FiB	4 पेन की कीमत 80 रुपये है। 10 पेन की कीमत _____ रुपये होगी। → Ans: (200)	Incorrect proportional scaling; arithmetic errors in cross-multiplication; misidentifying the unknown quantity	Providing a full sentence or expression as the answer instead of the required bare numeral; including units or currency symbols (e.g., "200 रुपये") when only a number is expected; omitting the required उत्तर: [संख्या] wrapper under HNaP.	FiB tasks are self-contained yet models struggle with multi-step proportional reasoning. Format errors are a secondary bottleneck as models over-explain rather than produce minimal numerical fill.
QNLI	Premise: 100 रुपये में 5 पेन... Hyp.: प्रत्येक पेन 20 रुपये... → Label: Entailment	Division error ($100 \div 5$); confusing <i>entailment</i> vs. <i>neutral</i> when quantities are consistent but not definitionally equivalent	Outputs raw relation label without required उत्तर: [संबंध] format under HNaP	No model exceeds 71.67% accuracy; requires precise mapping of arithmetic correctness onto logical relation—a dual failure point for all models
AWP	15 पेन 75 में खरीदे, 90 में बेचे... प्रतिशत लाभ? → Ans: 20%	Reports absolute profit (Rs. 15) instead of percentage; divides by wrong base (selling price instead of cost price)	Percentage symbol omitted or answer given in wrong unit (e.g., 15 रुपये instead of 20%)	Models excelling on FiB often fail here, revealing a gap between basic arithmetic and genuine word-problem comprehension

Language Sensitivity Under Hindi Prompting.

Hindi Native Prompting (HNaP) requires models to interpret instructions and produce outputs directly in Hindi while adhering to a fixed answer format. In contrast, Cross-Lingual Prompting (XLP) translates the task instructions into English while preserving the underlying numerical structure. Because the numerical content remains unchanged across these settings, performance differences between HNaP and XLP primarily reflect language realization and instruction-following difficulty rather than changes in task complexity.

Appendix D.1 provides instance-level evidence supporting this distinction. Specifically, the wf% (wrong-format) metric captures failures in output format compliance, while wc% (wrong-calculation) measures numerical errors among correctly formatted outputs. Together, these metrics establish an operational separation between language-related failures and computation-related failures.

Language Robustness Index (LRI). To quantify language sensitivity at the model level, we de-

Table 7: Detailed accuracy (%) scores for two independent evaluation runs across prompting strategies (HNaP, XLP, XCoT) and task categories (CA, DS, CQ, FiB, QNLI, AWP). The best results are **orange boldfaced** and second-best results are gray underlined

Models	CA	DS	CQ	FiB	QNLI	AWP	Avg.
Run 1							
<i>HNaP</i>							
Mathstral-7B	8.72	2.62	15.98	6.54	13.81	3.63	8.88
Llama-3.3-70B	59.57	24.42	53.77	42.15	22.53	46.50	41.49
DeepSeek-R1-Distill-Llama-70B	8.72	0.87	10.17	2.18	4.36	2.18	4.75
Qwen2.5-Math-72B	64.90	26.42	61.44	47.82	18.86	50.77	45.04
InternLM-Math-7B	51.90	17.42	48.44	32.82	12.86	41.97	34.23
GPT-4o	91.56	65.40	90.83	80.66	63.22	90.11	80.30
Gemini-2.0-Flash	95.92	82.84	92.29	85.02	59.58	<u>96.63</u>	85.38
GPT-5.2	<u>96.59</u>	<u>87.84</u>	<u>96.95</u>	<u>87.69</u>	<u>69.91</u>	<u>95.43</u>	<u>89.07</u>
Gemini-3.1-Pro	96.92	88.84	97.62	89.35	74.25	97.23	90.70
Claude-Opus-4.5	95.25	85.04	95.29	85.35	68.25	95.63	87.47
<i>XLP</i>							
Mathstral-7B	32.70	20.49	46.50	36.33	39.97	35.60	35.27
Llama-3.3-70B	90.11	68.02	91.56	84.29	58.13	90.11	80.37
DeepSeek-R1-Distill-Llama-70B	69.76	30.08	41.42	51.60	30.52	58.13	46.92
Qwen2.5-Math-72B	92.11	59.42	89.56	81.62	46.13	83.31	75.36
InternLM-Math-7B	80.11	46.42	75.56	67.62	30.13	65.91	60.96
GPT-4o	94.47	60.60	93.41	79.93	68.31	86.50	80.54
Gemini-2.0-Flash	98.10	73.25	96.65	93.01	61.77	<u>96.65</u>	86.24
GPT-5.2	<u>99.77</u>	<u>79.05</u>	<u>98.31</u>	<u>94.68</u>	<u>72.77</u>	<u>94.45</u>	<u>89.84</u>
Gemini-3.1-Pro	100.00	81.65	98.98	95.01	73.44	97.05	91.02
Claude-Opus-4.5	98.10	78.05	96.98	92.34	69.11	<u>96.65</u>	88.54
<i>XCoT</i>							
Mathstral-7B	39.97	21.80	51.58	31.25	40.69	42.15	37.91
Llama-3.3-70B	90.83	64.53	86.47	78.48	51.58	87.93	76.64
DeepSeek-R1-Distill-Llama-70B	66.13	20.49	45.05	39.97	26.16	37.06	39.81
Qwen2.5-Math-72B	90.83	56.53	87.47	70.14	41.91	85.26	72.02
InternLM-Math-7B	78.17	42.13	70.13	56.14	26.25	61.59	55.74
GPT-4o	93.74	55.81	92.29	72.67	63.22	81.39	76.52
Gemini-2.0-Flash	92.29	68.02	96.65	85.02	60.30	79.93	80.37
GPT-5.2	<u>95.63</u>	<u>78.22</u>	<u>94.98</u>	<u>90.69</u>	<u>73.63</u>	<u>93.33</u>	<u>87.75</u>
Gemini-3.1-Pro	96.29	79.22	<u>95.65</u>	91.02	73.97	<u>93.93</u>	88.35
Claude-Opus-4.5	93.96	75.62	93.31	87.69	71.97	94.33	86.15
Run 2							
<i>HNaP</i>							
Mathstral-7B	5.81	3.05	15.98	7.99	10.90	2.91	7.77
Llama-3.3-70B	66.13	27.90	55.95	45.78	15.98	43.60	42.56
DeepSeek-R1-Distill-Llama-70B	12.35	1.31	8.72	3.63	7.99	0.00	5.67
Qwen2.5-Math-72B	71.46	29.90	63.62	51.45	12.31	47.87	46.10
InternLM-Math-7B	58.46	20.90	50.62	36.45	6.31	39.07	35.30
GPT-4o	91.56	61.91	90.11	86.47	59.58	85.02	79.77
Gemini-2.0-Flash	95.92	78.92	85.74	89.38	58.13	<u>90.83</u>	83.15
GPT-5.2	<u>96.59</u>	<u>83.92</u>	<u>90.40</u>	<u>92.05</u>	<u>68.46</u>	<u>89.63</u>	<u>86.84</u>
Gemini-3.1-Pro	96.92	84.92	91.07	93.71	72.80	91.43	88.48
Claude-Opus-4.5	95.25	81.12	88.74	89.71	66.80	89.83	85.24
<i>XLP</i>							
Mathstral-7B	44.33	17.00	50.14	31.97	37.06	30.52	35.84
Llama-3.3-70B	88.65	66.27	91.56	82.84	57.41	88.65	79.56
DeepSeek-R1-Distill-Llama-70B	72.67	30.52	36.33	56.68	26.16	44.33	44.45
Qwen2.5-Math-72B	90.65	57.67	89.56	80.17	45.41	81.85	74.22
InternLM-Math-7B	78.65	44.67	75.56	66.17	29.41	64.45	59.82
GPT-4o	93.41	61.48	<u>94.47</u>	80.66	66.13	80.66	79.47
Gemini-2.0-Flash	97.37	76.74	92.29	94.47	63.94	<u>97.37</u>	87.03
GPT-5.2	<u>99.04</u>	<u>82.54</u>	<u>93.95</u>	<u>96.14</u>	<u>74.94</u>	<u>95.17</u>	<u>90.30</u>
Gemini-3.1-Pro	99.37	85.14	94.62	96.47	75.61	97.77	91.50
Claude-Opus-4.5	97.37	81.54	92.62	93.80	71.28	<u>97.37</u>	89.00
<i>XCoT</i>							
Mathstral-7B	38.50	17.88	52.32	29.80	32.70	31.97	33.86
Llama-3.3-70B	88.65	58.86	89.38	82.11	55.24	83.57	76.97
DeepSeek-R1-Distill-Llama-70B	59.58	18.75	59.58	36.33	21.07	33.43	38.79
Qwen2.5-Math-72B	88.65	50.86	90.38	73.77	45.57	80.90	71.69
InternLM-Math-7B	75.99	36.46	73.04	59.77	29.91	57.23	55.40
GPT-4o	93.74	56.68	92.29	72.67	63.22	79.18	76.30
Gemini-2.0-Flash	89.38	67.14	96.65	79.18	59.58	83.57	79.25
GPT-5.2	<u>92.72</u>	<u>77.34</u>	<u>94.98</u>	<u>84.85</u>	<u>72.91</u>	<u>96.97</u>	<u>86.63</u>
Gemini-3.1-Pro	<u>93.38</u>	78.34	<u>95.65</u>	85.18	73.25	<u>97.57</u>	87.23
Claude-Opus-4.5	91.05	74.74	93.31	81.85	71.25	97.97	85.03

fine the Language Robustness Index (LRI) as the difference between performance under Hindi Na-

tive Prompting and Cross-Lingual Prompting:

$$\text{LRI} = \text{Accuracy}_{\text{HNaP}} - \text{Accuracy}_{\text{XLP}} \quad (5)$$

Because both prompting conditions preserve identical numerical content, this index isolates the performance degradation associated with Hindi surface realization and instruction-following requirements.

As visualized in Figure 8, large negative LRI values indicate substantial performance degradation under Hindi-native prompting, suggesting that language realization and format adherence remain limiting factors. Conversely, values near zero indicate stable performance across prompting conditions, reflecting strong multilingual robustness.

Reasoning Residual Gap (RRG). To quantify residual reasoning limitations after controlling for language effects, we define the Reasoning Residual Gap (RRG) as the difference between human accuracy and model accuracy under Cross-Lingual Prompting:

$$\text{RRG} = \text{Human Accuracy} - \text{Accuracy}_{\text{XLP}} \quad (6)$$

Because XLP reduces language-related difficulty while preserving the numerical task structure, this metric reflects the remaining reasoning gap relative to human performance, as illustrated in Figure 8.

F Controlled Scaling & Distillation Analysis

We conduct a supplementary controlled-scaling study within a single model family. This analysis isolates the effects of parameter count from confounding factors such as architecture differences, training data composition, and distillation procedures.

Experimental Setup We evaluate the Qwen2.5 family—a multilingual lineage with consistent architecture and training protocols—across four contiguous scales: 7B, 14B, 32B, and 72B parameters. For distillation analysis, we include distilled variants at the 7B and 70B scales (Qwen2.5-Math-Distill), trained via standard knowledge distillation from the respective base models on English-dominant mathematical corpora. All models are evaluated zero-shot on HI-REASO under Hindi Native Prompting (HNaP) and Cross-Lingual Prompting (XLP), with temperature fixed at 1.0 and identical prompt templates.

Scaling Law Analysis As shown in Figure 9, fitting a log-linear model $\text{Accuracy} = \alpha \cdot \log(\text{Params}) + \beta$ to the HNaP results yields $R^2 \approx 0.96$, confirming that Hindi numerical reasoning follows predictable scaling dynamics within a controlled family. However, diminishing returns emerge beyond 30B parameters: the gain from 32B→72B (+13.2% HNaP) is markedly smaller than from 7B→32B (+23.3%). This empirically supports our claim that size alone is not the primary bottleneck for low-resource numerical reasoning.

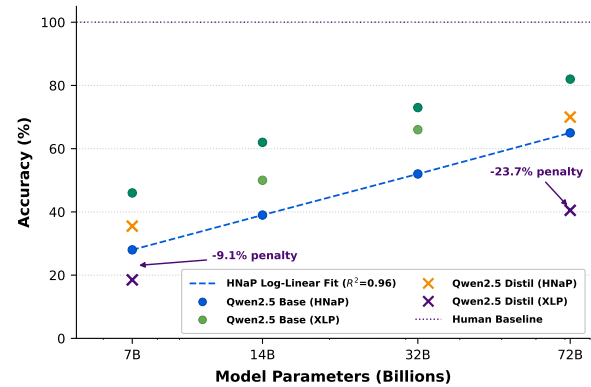


Figure 9: This figure presents a controlled scaling analysis within the Qwen2.5 family. The log-linear fit ($R^2 \approx 0.96$) confirms predictable scaling for Hindi numerical reasoning, while the pronounced distillation penalty (red/orange $\times$ markers) isolates knowledge compression as a distinct failure mode.

Distillation Penalty The distillation effect is both substantial and scale-dependent. At 7B, distillation reduces HNaP accuracy by 9.1 percentage points; at 70B/72B, this penalty more than doubles to 23.7 points. This pattern aligns with our main-paper observation that DeepSeek-R1-Distill-Llama-70B (14.8% HNaP) substantially underperforms Llama-3.3-70B (48.6% HNaP), but now isolates the effect from architectural or training-data confounders.

Connection to Error Analysis As shown in Appendix D.1 (Tables 4–5), distilled models exhibit sharply elevated wrong-format error rates (wf%) under HNaP—e.g., DeepSeek-R1-Distill-Llama-70B reaches wf% = 78.1% on HNaP versus 33.7% under XLP. This suggests that distillation, typically optimized on English reasoning trajectories, prunes token distributions critical for Hindi instruction-following and numeral syntax. The scaling results thus mechanistically explain why distilled models fail: compression sacrifices

low-resource language alignment in favor of high-resource reasoning patterns.

G Experimental Details

G.1 Hindi Native Prompting (HNaP).

As shown in Table 4 & Table 5, several models – most notably DeepSeek-R1-Distill-Llama-70B – exhibit extremely high wf%, indicating poor adherence to output constraints under native-language prompting. In such cases, wc% remains artificially low because few predictions qualify for reasoning evaluation. In contrast, GPT-4o and Gemini-2.0-Flash maintain strong format compliance but still incur substantial wc% on tasks such as QNLI, revealing reasoning failures despite correct formatting.

G.2 Cross-Lingual Prompting (XLP).

As reported in Table 4 & Table 5, XLP substantially improves format adherence across all models. For example, Mathstral-7B reduces wf% from HNaP to XLP on DS tasks, while Llama-3.3-70B consistently maintains wf% below 10%. However, wc% remains high for several models, indicating that improved formatting alone does not resolve deeper reasoning limitations.

G.3 Cross-Lingual Chain-of-Thought Prompting (XCoT)

Table 4 & Table 5 shows that XCoT generally reduces wc% by encouraging step-by-step reasoning, particularly for stronger models such as Gemini-2.0-Flash. However, verbose rationales often increase wf% for models with weaker instruction-following abilities, such as DeepSeek-R1-Distill-Llama-70B, and occasionally for GPT-4o. Llama-3.3-70B achieves a favorable balance, with low wf% and moderate wc%.

Together, Table 5 shows that formatting compliance and reasoning accuracy are not tightly coupled. High wf% indicates instruction-following failures, while high wc% reveals reasoning errors despite correct formatting. This underscores the need for disaggregated error analysis to diagnose model- and prompt-specific failure modes. This appendix provides supplementary material to support reproducibility and deeper inspection of our experimental setup. We include the prompt templates used for each prompting strat-

egy: Cross-Lingual Chain-of-Thought Prompting (XCoT), Cross-Lingual Prompting (XLP), and Hindi Native Prompting (HNaP). Additional examples from the HI-REASO dataset are presented in the task category box. Finally, Table 7 reports detailed results for each individual evaluation run.

Prompt for HINDI NATIVE PROMPTING (HNAP)

निर्देश:

आप एक गणित-विशेषज्ञ AI मॉडल हैं, जो हिंदी भाषा में पूर्णतः दक्ष हैं। आपका कार्य दिए गए गणितीय प्रश्न को चरणबद्ध रूप से हल करना और अंतिम उत्तर हिंदी में प्रस्तुत करना है।

(You are an AI model specializing in mathematics and fully proficient in Hindi. Your task is to solve the given mathematical problem step by step and provide the final answer in Hindi.)

Applicable Tasks: Tasks 1 (CA), 2 (DS), 4 (FiB), 6 (AWP)

कृपया निम्नलिखित प्रारूप का पालन करें:

- उत्तर: [] (Answer: [number])

#प्रश्न (Problem):

<question>

#समाधान (Solution):

<LLM Response>

Task 3: CQ

निर्देश:

आप एक गणित-विशेषज्ञ AI मॉडल हैं, जो हिंदी भाषा में दक्ष हैं। आपका कार्य दिए गए गणितीय प्रश्न का विश्लेषण करना और सही विकल्प का चयन करना है।

(Analyze the problem and choose the correct option.)

कृपया निम्नलिखित प्रारूप का पालन करें:

- उत्तर: [] (Correct option)

#प्रश्न (Problem):

<question>

#उत्तर 1: <option1>

#उत्तर 2: <option2>

#समाधान (Solution):

<LLM Response>

Task 5: QNLI

निर्देश:

आप एक गणित-विशेषज्ञ AI मॉडल हैं, जो हिंदी भाषा में दक्ष हैं। आपका कार्य दिए गए पूर्वकथन और अनुमान के बीच तार्किक-संख्यात्मक संबंध निर्धारित करना है।

(Determine the numerical-logical relationship between the premise and the hypothesis.)

कृपया निम्नलिखित प्रारूप का पालन करें:

- उत्तर: [] (Correct relation)

Options: Entailment / Neutral / Contradiction

#पूर्वकथन (Premise):

<premise>

#अनुमान (Hypothesis):

<hypothesis>

#समाधान (Solution):

<LLM Response>

Prompt for CROSS-LINGUAL PROMPTING (XLP)

You are a Math Expert AI model proficient in Hindi and English. Your task is to solve a mathematical reasoning problem provided in Hindi and deliver the solution in English.

Applicable Tasks: Tasks 1 (CA), 2 (DS), 4 (FiB), 6 (AWP)

Format:

- Translate the Hindi problem into English.
- Conclude with Answer: [num].

Problem:

<question>

Response:

<LLM Response>

Task 3 CQ

You are a Math Expert AI model proficient in Hindi and English. Your task is to solve a mathematical reasoning problem provided in Hindi and choose the correct option.

Format:

- Translate the Hindi question and options into English.
- Conclude with Answer: [Option] (Option 1 or Option 2).

Question:

<question>

Option 1: <option1>

Option 2: <option2>

Response:

<LLM Response>

Task 5 QNLI

You are a Math Expert AI model proficient in Hindi and English. Your task is to solve a Quantitative Natural Language Inference (QNLI) problem presented in Hindi and determine the relationship between the premise and the hypothesis.

Format:

- Translate the premise and hypothesis into English.
- Conclude with Answer: [Option].

Options: Entailment / Neutral / Contradiction

Premise:

<premise>

Hypothesis:

<hypothesis>

Response:

<LLM Response>

Prompt for CROSS-LINGUAL CHAIN-OF-THOUGHT (XCoT)

You are a Math Expert AI model proficient in Hindi and English. Your task is to solve a numerical reasoning problem provided in Hindi and deliver the solution in English.

Applicable Tasks: Tasks 1 (CA), 2 (DS), 4 (FiB), 6 (AWP)

Instructions:

- Translate the Hindi problem into English.
- Provide step-by-step reasoning in English.
- Conclude with Answer: [num].

Problem:

<question>

Response:

<LLM Response>

Task 3: CQ

Instructions:

- Translate the question and options into English.
- Perform step-by-step reasoning.
- Conclude with Answer: [Option].

Question:

<question>

Option 1: <option1>

Option 2: <option2>

Response:

<LLM Response>

Task 5: QNLI

Instructions:

- Translate the premise and hypothesis into English.
- Provide step-by-step reasoning.
- Conclude with Answer: [Option].

Options: Entailment / Neutral / Contradiction

Premise:

<premise>

Hypothesis:

<hypothesis>

Response:

<LLM Response>

Examples from HI-REASO by Task Category

Question:

एक सप्ताह में 7 दिन होते हैं। 3 सप्ताह में कुल कितने दिन होंगे?
(There are 7 days in a week. How many days are there in 3 weeks?)

Answer: 21 **Question:**

एक कार में 4 पहिए होते हैं। 12 कारों में कुल कितने पहिए होंगे?
(A car has 4 wheels. How many wheels are there in 12 cars?)

Answer: 48

DS**Question:**

12 रेडियन को डिग्री में बदलने पर मान कितना होगा?
(What is the value of 12 radians in degrees?)

Answer: 687.55° **Question:**

5 मीटर त्रिज्या वाले वृत्त का क्षेत्रफल कितना होगा?
(What is the area of a circle with radius 5 meters?)

Answer: 78.54

CQ**Question:**

संख्या 12345 क्या 54321 से छोटी है या बड़ी? (1) छोटी (2) बड़ी
(Is 12345 smaller or larger than 54321? (1) Smaller (2) Larger)

Answer: (1) **Question:**

एक पेड़ की ऊँचाई 8 मीटर है और दूसरे की 5 मीटर। कौन सा पेड़ अधिक ऊँचा है?
(One tree is 8 meters tall and another is 5 meters. Which is taller?)

Answer:

FiB**Question:**

4 पेन की कीमत 80 रुपये है। 10 पेन की कीमत _____ रुपये होगी।
(The price of 4 pens is 80 rupees. What is the price of 10 pens?)

Answer: 200

Question:

15 का दुगुना _____ होता है।
(The double of 15 is ____.)

Answer: 30

QNLI**Premise:**

राहुल ने 50 रुपये में एक किताब खरीदी।
(Rahul bought a book for 50 rupees.)

Hypothesis:

राहुल ने 50 रुपये में दो किताबें खरीदीं।
(Rahul bought two books for 50 rupees.)

Label: Contradiction

Premise:

अमित ने 100 रुपये में 5 पेन खरीदे।
(Amit bought 5 pens for 100 rupees.)

Hypothesis:

प्रत्येक पेन की कीमत 20 रुपये है।
(Each pen costs 20 rupees.)

Label: Entailment

AWP**Question:**

15 पेन 75 रुपये में खरीदे गए और 90 रुपये में बेच दिए गए। प्रतिशत लाभ कितना होगा?
(15 pens were bought for 75 rupees and sold for 90 rupees. What is the profit percentage?)

Answer: 20% **Question:**

800 रुपये में 4 किलोग्राम चावल मिलता है। 600 रुपये में कितने किलोग्राम चावल मिलेंगे?
(If 4 kg of rice costs 800 rupees, how much rice can be bought for 600 rupees?)

Answer: 3