IndicIFEval: A Benchmark for Verifiable Instruction-Following Evaluation in 14 Indic Languages

Anonymous ACL submission

Abstract

Instruction-following benchmarks remain predominantly English-centric, leaving a critical evaluation gap for the millions of Indic language speakers. We introduce IndicIFEval, a benchmark evaluating constrained generation of LLMs across 14 Indic languages using automatically verifiable, rule-based instructions. It comprises around 800 human-verified examples per language spread across two complementary subsets: IndicIFEval-Trans, translated prompts from *IFEval* (Zhou et al., 2023) carefully localized for Indic contexts, and IndicIFEval-Ground, synthetically generated instructions grounded in native Indic content. We conduct a comprehensive evaluation of major open-weight and proprietary models spanning both reasoning and non-reasoning models. While models maintain strong adherence to formatting constraints, they struggle significantly with lexical and cross-lingual tasks – and despite progress in high-resource languages, instruction-following across the broader Indic family lags significantly behind English. We release IndicIFEval and its evaluation scripts to support progress on multilingual constrained generation.

1 Introduction

Instruction following, the ability of a model to accurately interpret user intent and comply with specified instructions, is a critical aspect of LLM training and evaluation. This capability is commonly assessed using either human annotators (Karpinska et al., 2021; Chiang et al., 2024) or LLM-as-a-Judge approaches (Zheng et al., 2023; Li et al., 2023). However, human evaluation is costly and time-consuming, while LLM-based judges can be inconsistent and biased (Wang et al., 2024; Zheng et al., 2023). A complementary paradigm using automatically verifiable instructions (where adherence can be determined through simple rule-based checks) has therefore gained traction, as exemplified by the IFEval benchmark (Zhou et al., 2023). These include instructions such as requesting the presence of certain words, a fixed number of sentences, or specific formats (e.g., JSON), all checkable via automated scripts.

At a multilingual level, this evaluation gap becomes more pronounced, as LLMs are often significantly weaker in non-English languages – not just as evaluators (Hada et al., 2024; Fu and Liu, 2025) but also in text generation (Ahuja et al., 2023; Üstün et al., 2024). This raises a natural question: *In multilingual and low-resource settings, to what extent can LLMs at least follow simple, rule-based, verifiable constraints?*

We address this gap with IndicIFEval, a benchmark evaluating verifiable instruction-following across 14 Indic languages, a linguistically diverse family characterized by data scarcity, morphological richness, and distinct script systems, making it a meaningful testbed for both multilingual and low-resource evaluation. Our contributions are:

(1) IndicIFEval-Trans: Translated and localized English IFEval prompts, carefully filtered and manually verified by native speakers for cultural suitability and translation quality.

(2) IndicIFEval-Ground: Synthetically generated instructions grounded in native Indic topics and content, manually verified by native speakers. Unlike translated prompts, these reflect naturalistic constraints with more real-world contexts.

(3) Comprehensive multilingual evaluation: The first comprehensive multilingual evaluation of verifiable instruction-following spanning diverse model families, parameter sizes from 0.5B to 70B, both reasoning and non-reasoning configurations, and both open-weight and proprietary models.

(4) Benchmark extension framework: A comprehensive framework for extending verifiable instruction-following benchmarks to new languages with the availability of models tailored

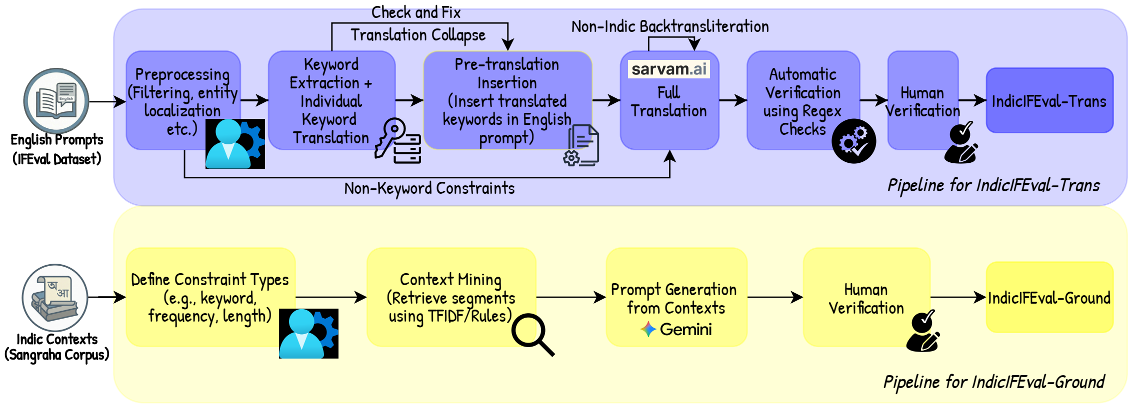

Figure 1: Overview of the dataset construction pipeline

for translation or general use, and corpora of local contexts. Together, IndicIFEval-Trans and IndicIFEval-Ground form around 800 human-verified examples per language, providing complementary perspectives on multilingual instruction-following: the former enabling direct comparison against English baselines, the latter grounding evaluation in more natural constraints that translation alone cannot capture.

2 Related Work

Several recent works have extended the verifiable instruction-following paradigm to multilingual contexts. M-IFEval (Dussolle et al., 2025) adapts the format to French, Japanese, and Spanish, highlighting that while models excel at formatting constraints, they struggle with script-specific tasks (e.g., "do not use Katakana"). Similarly, CL-IFEval (Ojewale et al., 2025) expands this slightly to include French, Spanish, Hindi, Arabic, and Yoruba. These works mainly employ translations.

More comprehensive benchmarks have recently been proposed. MaXIFE (Liu et al., 2025) introduces a dataset covering 23 languages, with 795 prompts and ∼1.7k constraint templates, resulting in over 18,000 possible combinations. It features a diverse taxonomy derived from real examples and uses both LLM-as-a-judge and accuracy-based evaluation. However, it relies on translation without localization. Kamath et al. (2025) craft new Hindi prompts with verifiable constraints through an in-house human annotation process guided by reference samples and Indian cultural themes.

Marco-Bench-MIF (Zeng et al., 2025) evaluates 30 languages with a specific focus on the distinction between translation and localization. They identify that machine-translated (MT) data underestimates model accuracy by 7–22% compared to localized data, largely due to implicit English constraints (like capitalization) that become invalid in translation. While this offers deep insights into localization, it does not explore creating new prompts from grounded contexts. Further, it does not compare performance between reasoning and non-reasoning modes, leaving a gap in understanding how reasoning affects instruction adherence.

3 Dataset Construction

In this section we detail the process of obtaining the translations based on the English IFEval dataset (Zhou et al., 2023) and LLM-based generations grounded in Indic contexts from `sangraha-verified` (Khan et al., 2024). An overview of the process is illustrated in Figure 1.

3.1 IndicIFEval-Trans

Preprocessing First, we manually inspect IFEval prompts for usability in non-English contexts. Wherever possible, we localize them with Indic-relevant themes through simple named-entity substitutions, such as replacing 'President of USA' with 'Prime Minister of India'. For constraints requiring explicit English responses, such as capitalization, we prepend "Respond in English." Second, we identify cases prone to problematic translation, including abbreviations that may not transfer well, translationese, tense changes, and incomplete MT outputs. We address most cases by replacing problematic terms with more translatable alternatives, while dropping a small number of difficult cases. This process yields 490 examples from the origi-

nal 541. Further details are provided in Appendix B.

Keyword Extraction & Translation We identify the prompts with keyword constraints and extract the keywords. The keywords are first translated individually. The ones that shouldn't be translated (e.g., prompts where the response language is in another language), are also translated, but they are stored separately.

Pre-translation Insertion Then, we insert the keyword translation into the respective English prompt in place of the English keywords. We call this the 'pre-translation'.

Full Translation Next, we translate the pre-translated prompts to the target language. Note that for non-keyword constraints, we translate them directly without modifications. We note that we preserve the non-English keywords as highlighted earlier. This is done by placing the keyword automatically back to its original source after translation. We use Sarvam-Translate (Sarvam, 2025) for the translations unless explicitly stated otherwise. This model is particularly suited for our pipeline as it delivers dedicated translation capabilities for 22 Indic languages while reliably handling structured long-form text.

Automatic Verification Finally, using regular expressions, we check that the constraints are described properly in the translations. We manually fix these errors in formatting wherever necessary. We discuss this in detail in Appendix B.

3.2 IndicIFEval-Ground

While translation provides a strong baseline for multilingual evaluation, it often lacks cultural grounding and may result in text that may be grammatically correct but constraints that are unnatural.

We observe that this unnaturalness is especially true for keyword and length constraints. Required keywords are often arbitrary translations of English terms, which may not naturally occur in the target language's context. Further, certain keyword constraints (e.g., "Start the response with X") are often grammatically infeasible or highly unnatural in Indic languages due to their distinct word order. As for length constraints (e.g., "Respond in X paragraphs" or "Respond in X sentences"), these may also not translate 1:1 to other languages. To address this, we employ a context-grounded synthetic data generation pipeline. The process involves the following steps:

(1) Define Constraint Types: We focus on a subset of six constraint types: Keyword Inclusion, Keyword Prohibition, Keyword Frequency, Paragraph Count, Sentence Count, and First Word, as these were identified to be the most problematic while translating from English.

(2) Context Mining: For valid keywords, we employ TFIDF-based sorting to extract top keywords that mimic common, real-world keywords people may use. Then, we retrieve text segments that match the defined constraint (e.g., contains five sentences, contains the keyword X two times, etc.) by looping over all combinations with an upper limit of 5 for frequency.

(3) Prompt Generation from Contexts: We use Gemini-2.5-Pro (Comanici et al., 2025) to generate a user prompt based on the provided context that requires satisfying the given constraint, with an explicit mention of the defined constraint. We also include a randomly chosen instruction type from a list of tasks to diversify the generated prompts.

3.3 Human Verification

To ensure the validity and quality of the dataset, we implement a human-in-the-loop verification process for both IndicIFEval-Trans and IndicIFEval-Ground subsets. We employ qualified, native speakers for each of the target languages as the annotators. For prompt verification, we assess the semantic validity of the prompt rather than strict grammatical perfection. Finally, we drop the prompts that are marked as incorrect - we do not perform any modifications to the prompts if they are incorrectly translated. Further details on the annotation guidelines along with the final dataset statistics in the Appendix C.

4 Evaluation

4.1 Models

We cover a diverse suite of models, spanning open-weights and proprietary architectures, across varying parameter sizes (from 1B to 70B+).

Models Evaluated. We evaluate a diverse set of open-weight and proprietary models spanning multiple families, parameter scales, and configurations. The open-weight models include Llama-3.1 (8B, 70B), Llama-3.2 (3B), Llama-3.3 (70B), and Llama-4 (17B-MoE-16E) (Grattafiori et al., 2024; Meta, 2024); Gemma 3 (1B, 4B, 12B, 27B) and Gemma 4 (E2B, E4B, 26B-A4b, 31B) (Gemma, 2025, 2026); Aya-Expanse (8B, 32B) (Dang et al., 2024); Qwen 3 (0.6B, 1.7B, 4B, 8B, 14B, 32B)

Model	as	bn	gu	hi	kn	ml	mr	ne	or	pa	sa	ta	te	ur	en	Avg
gemini/gemini-3-flash	90.0	90.0	89.7	89.7	86.3	84.7	88.8	89.7	86.9	89.4	79.4	85.7	89.4	85.4	96.3	87.5
gemini/gemini-3-pro	89.4	92.5	91.3	91.0	91.3	86.9	91.3	92.2	86.0	93.2	86.3	89.7	90.3	89.1	95.3	90.0
openai/gpt-5-2025-08-07	88.8	90.7	89.4	92.8	89.1	90.0	89.7	91.0	89.1	90.3	80.4	86.9	88.5	87.5	95.0	88.9
openai/gpt-5-mini-2025-08-07	77.9	86.0	84.4	88.8	72.9	76.3	81.6	82.9	75.4	84.1	70.1	75.1	80.1	81.3	91.9	79.8
CohereLabs/aya-expanse-8b	19.0	24.0	17.4	53.6	20.2	21.8	24.3	20.2	19.0	18.7	21.2	24.9	18.4	38.9	70.4	24.4
CohereLabs/aya-expanse-32b	33.0	42.7	33.0	70.4	30.8	39.9	32.7	37.7	35.5	32.1	29.9	41.4	26.5	57.6	77.3	38.8
Qwen/Qwen3-0.6B	23.7	30.2	23.7	30.5	28.7	23.4	27.7	25.2	22.7	26.5	24.0	21.5	25.9	28.0	65.4	25.8
Qwen/Qwen3-1.7B	32.4	43.3	35.5	48.3	38.3	35.5	39.2	37.7	41.1	34.0	28.7	31.8	34.9	44.9	75.4	37.5
Qwen/Qwen3-4B	54.2	62.6	54.2	66.4	54.2	48.0	57.9	58.9	47.4	46.7	48.3	48.6	53.3	59.8	85.0	54.3
Qwen/Qwen3-8B	59.5	67.3	63.9	74.1	67.9	61.7	66.7	64.8	52.6	57.6	54.5	60.8	62.0	64.8	88.2	62.7
Qwen/Qwen3-14B	69.2	72.6	72.9	78.8	75.1	69.5	70.4	72.3	74.5	70.1	64.5	65.4	69.8	70.4	87.8	71.1
Qwen/Qwen3-32B	71.7	73.8	74.8	78.5	74.1	73.5	75.7	73.5	70.4	67.6	64.8	73.2	72.0	75.4	87.2	72.8
google/gemma-4-E2B-it	58.3	74.1	71.0	77.6	65.1	63.5	68.8	69.8	48.3	59.5	46.1	66.4	68.2	67.9	88.0	64.6
google/gemma-4-E4B-it	70.7	80.1	79.1	81.9	77.3	73.8	77.3	76.3	63.2	72.3	58.6	75.4	71.3	78.8	90.5	74.0
google/gemma-4-26B-it	78.8	85.0	85.4	86.6	79.4	78.8	81.9	81.0	73.5	79.8	66.7	80.4	81.3	78.8	91.0	79.8
google/gemma-4-31B-it	81.3	87.2	85.4	90.0	85.0	82.5	85.4	84.4	81.6	87.2	74.8	82.9	84.4	85.0	93.0	84.1
meta-llama/Llama-3.2-1B-Instruct	16.5	17.8	13.4	27.1	15.9	14.0	22.1	22.4	18.1	16.5	20.6	15.9	15.6	20.6	54.2	18.3
meta-llama/Llama-3.2-3B-Instruct	23.4	34.9	22.7	42.7	23.7	17.8	30.8	29.0	17.1	25.9	18.4	22.1	28.4	39.2	77.6	26.9
meta-llama/Llama-3.1-8B-Instruct	38.9	44.9	29.0	53.6	39.2	31.8	41.7	45.5	28.0	35.8	32.4	32.1	33.0	49.2	81.6	38.2
meta-llama/Llama-4-17B-Instruct	73.2	76.6	77.0	77.3	75.1	73.5	74.1	75.1	66.4	76.0	52.0	69.5	72.0	76.6	89.7	72.5
meta-llama/Llama-3.1-70B-Instruct	60.4	67.0	55.5	71.0	59.5	55.5	60.8	64.8	52.6	59.8	51.1	54.5	61.4	74.1	88.5	60.6
meta-llama/Llama-3.3-70B-Instruct	72.0	77.0	73.2	79.4	72.0	66.7	74.8	74.8	63.5	70.1	67.3	69.2	72.3	76.3	91.6	72.0
sarvamai/sarvam-30b	41.4	43.9	46.7	49.2	44.5	38.6	48.9	42.7	40.2	40.8	34.9	37.1	38.9	44.9	69.2	42.4
sarvamai/sarvam-105b	53.3	58.9	59.5	62.3	57.9	52.3	57.3	57.6	53.3	51.1	46.1	50.2	52.6	59.2	76.3	55.1
Avg	56.9	64.0	60.0	69.3	58.9	56.6	61.6	61.3	53.2	57.2	50.6	57.4	58.4	63.8	83.1	59.2

Table 1: Overall Results for IndicIFEval-Trans (avg calculated only for Indic languages)

(Yang et al., 2025); and Sarvam-30B and 105B (Sarvam Foundation Models Team, 2026). We additionally evaluate proprietary frontier models, including GPT-5 and GPT-5-mini (Singh et al., 2025), as well as Gemini-3-Pro and Gemini-3-Flash (Google, 2025).

4.2 Metrics

Following IFEval (Zhou et al., 2023), we report `prompt-level` and `instruction-level` accuracy. The former measures the percentage of prompts where *all* constraints are satisfied, while the latter measures the percentage of *individual instructions* satisfied. We report `loose` scores to account for minor response issues like formatting.

In our experiments, we observe that `instruction-level` accuracy scores are higher than `prompt-level`, however, the overall trends themselves are consistent, and hence, we focus primarily on prompt-level accuracy. In constraint-wise comparisons, accuracy is computed per instruction and then averaged across instructions, so the choice of reporting prompt-level versus instruction-level accuracy does not change the scores in these analyses.

4.3 Evaluation

For evaluation, the original IFEval scripts use NLTK (Bird and Loper, 2004) to verify structural constraints such as word, sentence, and paragraph counts. To reliably evaluate these constraints across our target languages, we integrate the Indic NLP Library (Kunchukuttan, 2020) into the verification pipeline. This provides sentence segmentation and word tokenization optimized for Indic scripts, ensuring that subsequent scoring remains precise. To standardize the generation process across our diverse model suite, all evaluations are orchestrated using the Language Model Evaluation Harness (Gao et al., 2024). We enforce greedy decoding during all inference runs.

5 Results and Analysis

5.1 IndicIFEval-Trans

To analyze cross-lingual trends, we require a strictly parallel corpus. We therefore curate a common subset of prompts that are marked as correct by annotators across languages, amounting to 321 examples per language. The prompt-level loose accuracy scores are presented in Table 1.

Across models, performance on English prompts (`en`) is consistently higher than the average over Indic languages (`Avg`), indicating a persistent cross-lingual gap in following verifiable instructions. This gap generally narrows for larger and more multilingual-oriented models, but remains pronounced for smaller and less multilingual models.

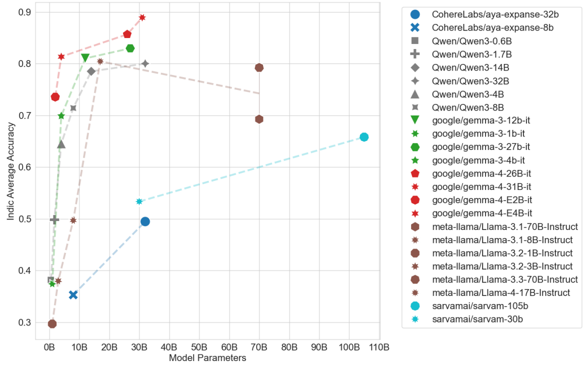

Figure 2: Effect of Increasing Model Parameters. The higher the better – The Gemma-4-31B-IT model performs the best, followed by Llama-4-Scout-17B-16E.

Effect of Model Parameters We find that increasing model parameters generally improves performance. Gemma-3-31B-IT wins overall followed by Gemma-3-27B-IT and Llama-4-Scout-17B-16E-Instruct. Qwen3-1.7B performs almost as well as the much larger Aya-Expanse-32B, possibly because of distilled thinking capabilities. At the smallest scale, Llama-3.2-1B-Instruct performs the poorest, and Gemma-3-1B-IT offers only marginal improvements, indicating that the *advantage of robust multilingual pre-training*, when comparing Gemma to Llama, *only materializes at larger scales* where the model has sufficient capacity to leverage it. Notably, Gemma 4 displays the best performance relative to its parameter count. Furthermore, performance plateaus around the 12B to 14B parameter range with limited gains thereafter (< 5), suggesting verifiable instruction adherence requires a minimum baseline of parameter capacity.

Note: We observe that the Indic performance is generally lower than the English baseline. Hence, to quantify how small or large this gap is to achieve parity, we present the subsequent analyses by viewing the disparity between the Indic language score and the English score('Δ'; ranging -1 to 1).
The lower the absolute gap, the better.

Model Family Performance The *Gemma family* consistently exhibits the *smallest performance gap with English* across all languages. While the Gemma 3 models maintained a gap of $\Delta < 0.15$ for languages like Hindi and Bengali, the Gemma 4 models further narrow this disparity to $\Delta < 0.07$ for the same languages at the 26B scale. This suggests the Gemma training objective is highly robust to cross-lingual inputs, especially considering its largest variant is less than half the size of the 70B models evaluated. The *Sarvam family* similarly demonstrates a remarkably low English-Indic disparity across the evaluated languages, confirming the effectiveness of targeted Indic post-training for structural instruction adherence.

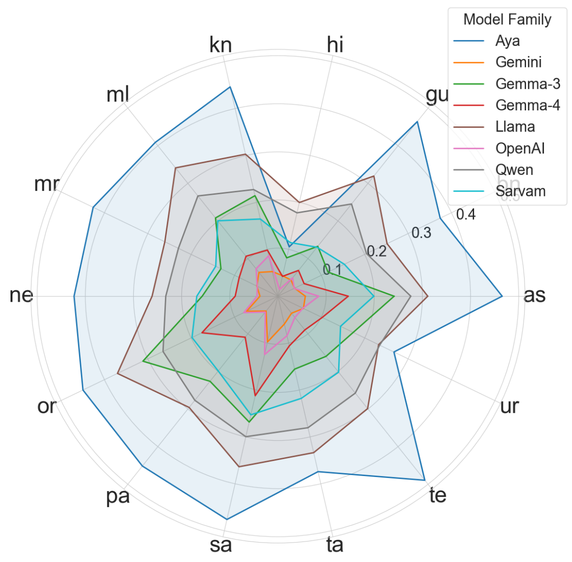

Figure 3: Indic - English disparity comparing Language vs Model Family. The lower the better – the Aya family exhibits the highest gap in general, but performs moderately well for Hindi, Bengali, Urdu, and Tamil.

The Qwen and Llama families occupy a competitive middle tier, with Qwen frequently outperforming Llama by maintaining slightly lower disparities in languages like Bengali and Gujarati. Conversely, the *Aya family* displays the highest degradation, with gap scores exceeding 0.40 for most low-resource languages, indicating a *significant struggle to transfer instruction-following precision despite its multilingual focus*. Notably, Hindi remains the most robust language across all families.

Individual Constraint Capabilities To provide a more granular understanding of model capabilities across different constraints, we examine the average performance for each constraint type. The standard IFEval framework consists of 25 distinct instruction constraints. Analyzing these constraints individually would be difficult and may yield fragmented insights. Hence, we aggregate these 25 constraints into five high-level semantic categories: ENGLISH, KEYWORD, FORMAT, LENGTH, and LANGUAGE to find broader patterns. The specific mapping of individual IFEval instruc-

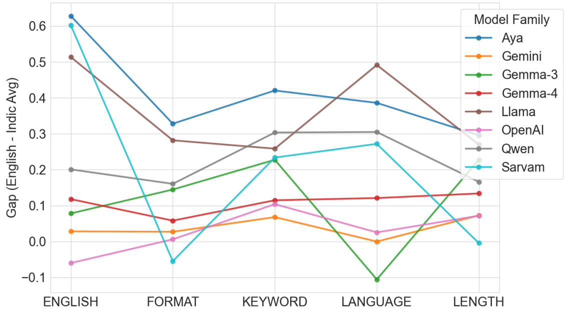

Figure 4: Indic - English gap comparing Model Family vs Instruction Category. Lower the better – the Llama family and Gemma family perform the best and worst respectively on LANGUAGE constraint within the family.

tions to these five categories is detailed in Table 3.

We observe that models consistently demonstrate *strong adherence to FORMAT constraints*. The Gemma and Qwen families achieve the highest performance in this category, followed closely by the Llama family. The Gemma-4 models further distinguish themselves by maintaining a consistently low performance gap across all five semantic categories, resisting the typical fluctuations seen in other architectures. Conversely, the Sarvam models exhibit highly polarized capabilities. They approach parity for structural constraints like FORMAT and LENGTH while experiencing severe degradation when handling ENGLISH constraints. Finally, the Aya model family exhibits significantly lower performance across all categories compared to other evaluated families (see Figure 4).

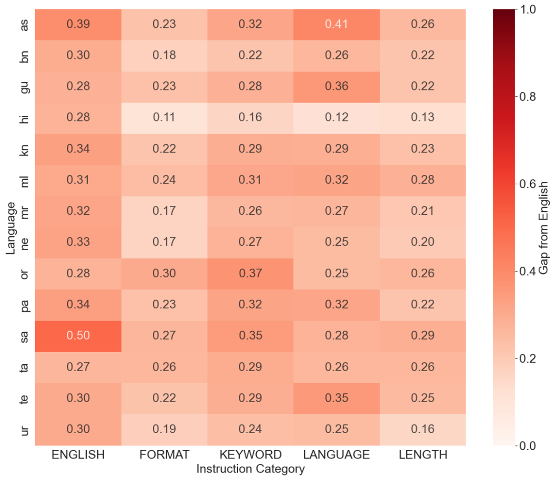

Figure 5: Indic - English disparity comparing Instruction Category vs Language. Lower the better – Hindi ('hi') is visually the lightest (lowest Δ) across all categories horizontally and darkest (highest Δ) across all languages vertically.

Moving to individual languages (refer Figure 5), we find that among all evaluated languages, Hindi exhibits the minimum Δ across the FORMAT, LANGUAGE, and LENGTH constraints. This indicates that high-resource Indic languages like Hindi are nearing parity with English in structural instruction following. The most significant performance degradation relative to English is observed when models are prompted in Sanskrit, particularly when the constraints require an English response.

Keyword Inclusion and Exclusion Within the KEYWORD category, we observe a strong performance disparity between negative constraints (Exclusion) and positive constraints (Inclusion). Across all model families, models demonstrate high proficiency in avoiding specific keywords when instructed. Conversely, models significantly struggle with including keywords, be it at least once or a certain number of times.

This gap suggests that while *models possess the semantic understanding to suppress concepts* (negative constraints), they *face challenges in generating the precise lexical surface forms required for inclusion* (positive constraints).

Cross-lingual Instruction Following Given that models demonstrate high proficiency in English generation when prompted in English, we analyze their performance when the same English output is requested via Indic prompts (i.e., comparing the ENGLISH column in Figure 4 and Figure 5 to other columns). Interestingly, we find that *shifting the prompt language introduces a performance gap, despite the general higher English capabilities*. Even high-resource languages like Hindi exhibit their maximum performance gap in this category ($\Delta = 0.28$) overall, significantly higher than all other constraints where $\Delta = 0.11 - 0.16$ (Figure 5). The Llama and Aya families suffer massive degradation ($\Delta > 0.50$). Upon manual inspection of the generations, we find that *models fail in such cross-lingual English instruction following* either when the *model continues to respond in the prompt language (Indic)*, thereby failing the English response constraint, or *respond in English but do not follow the capitalization constraints*.

The Gemma family effectively closes this gap having $\Delta = 0.08$ for LANGUAGE (Figure 4). To our surprise, we find that the Gemma family performs worse on the ENGLISH constraint than on the LANGUAGE constraint, with the ENGLISH cat-

Model	as	bn	gu	hi	kn	ml	mr	ne	or	pa	sa	ta	te	ur	Avg
gemini/gemini-3-flash	64.6	67.0	60.7	61.9	58.1	56.2	62.9	66.9	62.8	64.5	55.6	64.9	64.3	59.6	62.1
gemini/gemini-3-pro	62.3	66.0	61.0	58.9	61.0	59.4	62.0	64.2	67.9	65.5	50.9	66.0	70.4	60.2	62.6
openai/gpt-5-2025-08-07	73.2	72.6	77.5	77.9	73.6	70.6	78.0	72.4	80.4	78.5	78.3	78.1	78.4	72.7	75.9
openai/gpt-5-mini-2025-08-07	75.5	71.2	74.5	75.5	71.5	68.2	81.7	75.7	78.8	77.0	79.5	79.1	78.7	73.6	75.8
CohereLabs/aya-expanse-8b	38.1	46.0	38.2	48.4	41.1	43.2	42.0	40.5	39.1	43.8	25.6	52.1	39.5	40.4	41.3
CohereLabs/aya-expanse-32b	59.9	56.2	46.1	54.9	45.8	55.5	53.1	50.7	47.8	42.5	36.0	58.4	48.0	42.4	49.8
Qwen/Qwen3-0.6B	48.6	53.5	50.7	56.9	47.5	42.7	56.0	54.8	32.3	38.6	43.5	49.3	51.2	47.8	48.1
Qwen/Qwen3-1.7B	59.9	56.7	51.5	58.4	48.1	45.6	62.6	61.9	44.8	37.9	53.9	55.1	49.9	54.0	52.9
Qwen/Qwen3-4B	71.2	69.4	61.3	63.7	58.1	55.2	62.6	62.8	62.2	48.7	67.0	61.2	64.8	59.9	62.0
Qwen/Qwen3-8B	73.2	70.2	65.2	64.3	67.5	60.9	65.4	73.0	68.2	63.3	62.8	71.2	67.2	60.2	66.6
Qwen/Qwen3-14B	74.3	72.1	71.3	73.1	69.2	70.3	69.7	75.7	67.4	67.5	64.3	76.7	68.5	60.2	70.0
Qwen/Qwen3-32B	73.2	70.2	64.2	62.4	61.3	61.5	66.6	70.4	69.3	64.8	61.3	70.0	71.2	57.6	66.0
google/gemma-4-E2B-it	52.1	53.8	54.9	53.2	51.0	52.1	56.6	55.4	63.3	56.5	58.0	59.5	60.0	52.5	55.6
google/gemma-4-E4B-it	58.8	56.7	56.5	56.5	52.5	50.0	60.6	64.2	56.8	58.0	62.5	64.2	61.9	55.5	58.2
google/gemma-4-26B-it	52.9	58.7	55.4	56.2	53.4	55.0	58.3	62.5	52.7	57.7	57.7	54.9	62.1	62.6	57.1
google/gemma-4-31B-it	57.2	59.7	52.2	60.0	53.4	54.7	58.9	65.7	55.4	61.1	48.2	64.0	59.7	59.4	57.8
meta-llama/Llama-3.2-1B-Instruct	42.4	63.6	65.0	60.8	42.2	47.1	53.7	57.8	31.5	46.7	50.9	55.1	61.6	59.4	52.7
meta-llama/Llama-3.2-3B-Instruct	58.0	71.4	53.6	64.8	64.8	56.8	70.9	63.0	48.4	59.7	44.6	71.6	70.1	61.7	61.4
meta-llama/Llama-3.1-8B-Instruct	62.6	68.7	63.4	65.2	62.8	62.8	61.1	71.3	56.2	62.6	65.2	68.8	70.7	56.4	64.1
meta-llama/Llama-4-17B-Instruct	70.4	71.6	72.9	65.9	70.4	63.8	79.1	75.7	73.1	67.0	70.8	71.4	74.4	67.7	71.0
sarvamai/sarvam-30b	46.7	46.2	48.8	49.7	45.5	40.4	52.0	49.6	48.1	50.1	41.7	43.7	51.5	41.8	46.8
sarvamai/sarvam-105b	49.0	45.0	50.1	50.1	42.5	39.6	57.4	48.4	48.1	50.1	41.1	47.9	49.3	49.0	47.7
Avg	60.0	61.8	59.3	60.9	56.2	55.2	62.2	62.8	56.3	57.4	55.5	62.8	62.5	56.4	59.2

Table 2: Overall Results for INDICIFEVAL-GROUND

egory $\Delta = -0.11$, the only case it happens to be negative. This negative difference signifies that *prompting in English* and instructing to respond in another language is *worse* than *prompting in the target language* and instructing to respond in another language. However, a qualitative analysis reveals that *Gemma models frequently provide translations or transliterations in parentheses for readability when the prompt is in English*. While helpful for human readers, this behavior *inadvertently violates strict language constraints*, thereby reducing the quantitative performance.

These discrepancies suggests that for many model families, cross-lingual instruction-following capabilities lag significantly behind their primary in-language capabilities, revealing a brittleness in how they ground Indic instructions to English outputs.

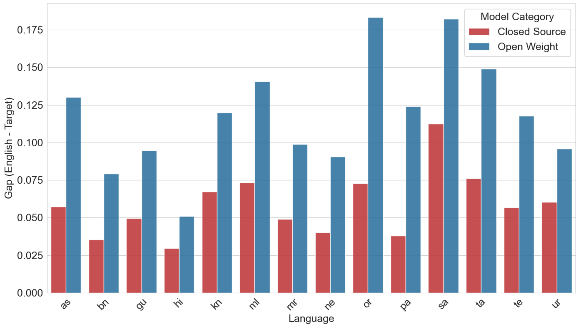

Figure 6: Indic vs English Disparity comparing Open-Weight and Proprietary. Lower the better – Hindi ('hi') exhibits the lowest gap with English in both proprietary (red) and open-weight settings (blue).

Open-Weight vs Proprietary For a better comparison of open-weight and proprietary models, we take the top two performing models from open-weight (i.e., Gemma-4-31B-IT and Llama-3.3-70B-Instruct) so as to not skew the results. We view the English vs Indic disparity (i.e., Δ) averaging across the two models selected for open-weight and four models in proprietary. We find that proprietary models consistently maintain a significantly lower Δ across all evaluated languages. While *open-weight models exhibit substantial variance depending on the language resource availability* (peaking at a disparity of $\Delta \approx 0.16$ for languages like Sanskrit and Odia), *their ability to close the gap is comparable to proprietary models in high-resource scenarios*. Notably, for Hindi, the open-weight Δ narrows to its minimum ($\Delta \approx 0.05$), closely approaching the proprietary baseline.

However, we note that when viewing the absolute accuracy scores (see Table 1), *there is still a 5+ score margin between the open-weight and proprietary models*, for verifiable constraints. Larger parameter sizes could potentially close this absolute performance gap, however, this is currently not visible, given that the best performing open-weight model is Gemma-4 at only 31B.

Thinking vs Non-Thinking Modes We choose the Qwen3 and Gemma 4 models as they both support thinking and non-thinking modes, allowing for a controlled ablation across varying parameters. As illustrated in Figure 7, *enabling the "Thinking" mode consistently narrows the performance disparity between English and Indic languages* across nearly all model scales, indicating that the additional thinking cost allows the model to better bridge the gap. At the lowest scale, the Qwen3-0.6B model exhibits a negligible improvement (an almost flat slope from 0.36 to 0.35), suggesting that sub-billion parameter models may lack the capacity to effectively leverage thinking or cross-lingual transfer.

Further, the roughly parallel trajectories (similar slope) of the thinking and non-thinking configurations demonstrate that the *performance benefit derived from the thinking mechanism remains constant and independent of the overall parameter count*. The Gemma-4-26B model aligns with this consistent slope by reducing its gap from 0.12 to 0.07. However, the Gemma-4-31B model displays a flat trajectory at 0.06, implying that thinking yields diminishing returns once a model approaches near parity with English.

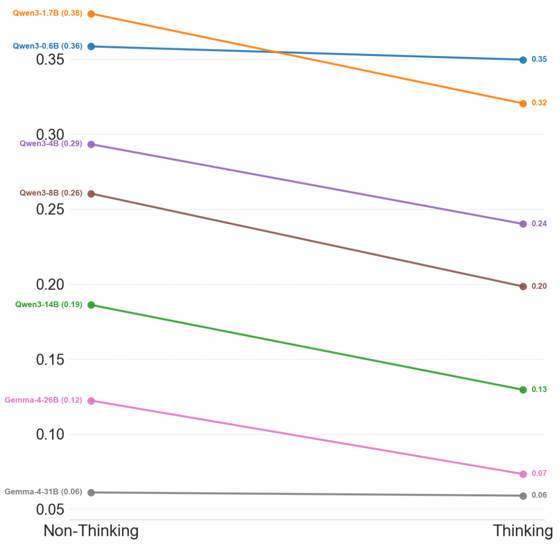

Figure 7: Indic vs English Disparity comparing Thinking and Non-Thinking Modes. A steeper negative slope indicates that thinking more effectively narrows the performance gap between English and Indic languages.

5.2 IndicIFEval-Ground

IndicIFEval-Trans, being parallel, allows for a direct comparison across languages. However, it relies on the assumption that translated prompts preserve the naturalness of the original instruction and previous research show that translations can be biased towards the original English-contexts (Zeng et al., 2025). We report the evaluation results (loose prompt-following acc.) in Table 2.

It is important to note that the results presented in this section are not directly comparable to the IndicIFEval-Trans scores discussed previously. The prompts in IndicIFEval-Ground differ significantly in complexity, topic, and structure. Consequently, there is no direct English baseline for these specific prompts; We focus exclusively on the absolute performance of models.

Further, the scores should not be viewed as direct comparison among Indic languages themselves as the prompts are entirely different for each language. This prompt diversity is necessary to capture the structural and cultural variance of Indic languages, which span multiple language families, and utilize different word orders.

6 Conclusion

We introduce IndicIFEval to benchmark verifiable instructions for 14 Indic languages. By pairing two complementary datasets with parallel prompts in IndicIFEval-Trans and natural constraints in IndicIFEval-Ground, this benchmark provides a comprehensive framework for assessing instruction following in multilingual and low-resource settings. Our evaluations reveal a persistent performance gap between English and Indic languages even for simple, rule-based instructions. High-resource languages like Hindi consistently exhibit a narrower gap and approach English parity. Among open-weight architectures, the Gemma family demonstrates the highest cross-lingual robustness, whereas proprietary models maintain a distinct overall advantage across the broader language spectrum. Models generally adhere to structural formatting instructions but face substantial difficulties with precise lexical generation, such as keyword inclusion, and cross-lingual tasks. Increasing model capacity and employing thinking modes consistently reduce the performance gap between English and Indic languages. Future work could focus on incorporating more complex and linguistically motivated constraints, and exploring alternative data creation strategies to enhance cultural grounding.

7 Limitations

Our grounded benchmark, IndicIFEval-Ground, explores only a subset of possible instruction constraint types and is therefore not exhaustive. In particular, we restrict our study to the 25 constraint categories defined in the original IFEval benchmark, whereas future work could incorporate more fine-grained and linguistically motivated constraints, such as romanization requirements, explicit specification of numeral formats (e.g., digits versus words), and system-of-units constraints (e.g., Indian vs International numeral systems, Imperial vs Metric system). In addition, our annotations were produced by a single annotator due to resource constraints; incorporating multiple annotators in future work would enable stronger verification through inter-annotator agreement and improve annotation reliability. Our exploration of reasoning capabilities is restricted to the Qwen and Gemma-4 models, leaving the evaluation of other models with thinking mechanisms for future work.

8 Ethical Considerations

Our dataset construction process relied on human verification performed by in-house annotators who are native speakers of the evaluated target languages. All participating annotators were compensated fairly and in strict accordance with standard organizational norms for their time and effort. Furthermore, we ensured that the entire data collection and human evaluation pipeline adhered to established ethical guidelines for NLP research.

References

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. 2023. MEGA: Multilingual evaluation of generative AI. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267, Singapore. Association for Computational Linguistics.

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

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E Gonzalez, and 1 others. 2024. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, and 1 others. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *arXiv preprint arXiv:2412.04261*.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*.

Antoine Dussolle, Andrea Cardeña Díaz, Shota Sato, and Peter Devine. 2025. M-IFEval: Multilingual instruction-following evaluation. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 6161–6176, Albuquerque, New Mexico. Association for Computational Linguistics.

Xiyan Fu and Wei Liu. 2025. How reliable is multilingual LLM-as-a-judge? In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 11040–11053, Suzhou, China. Association for Computational Linguistics.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2024. The language model evaluation harness.

Gemma. 2025. Gemma 3 technical report.

Gemma. 2026. Gemma 4: Byte for byte, the most capable open models.

Google. 2025. A new era of intelligence with gemini 3.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Rishav Hada, Varun Gumma, Adrian de Wynter, Harshita Diddee, Mohamed Ahmed, Monojit Choudhury, Kalika Bali, and Sunayana Sitaram. 2024. Are large language model-based evaluators the solution

to scaling up multilingual evaluation? In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1051–1070, St. Julian's, Malta. Association for Computational Linguistics.

Anusha Kamath, Kanishk Singla, Rakesh Paul, Raviraj Bhuminand Joshi, Utkarsh Vaidya, Sanjay Singh Chauhan, and Niranjan Wartikar. 2025. Benchmarking hindi llms: A new suite of datasets and a comparative analysis. In *Proceedings of the 1st Workshop on Benchmarks, Harmonization, Annotation, and Standardization for Human-Centric AI in Indian Languages (BHASHA 2025)*, pages 52–68.

Marzena Karpinska, Nader Akoury, and Mohit Iyyer. 2021. The perils of using Mechanical Turk to evaluate open-ended text generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1265–1285, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mohammed Safi Ur Rahman Khan, Priyam Mehta, Ananth Sankar, Umashankar Kumaravelan, Sumanth Doddapaneni, Suriyaprasaad G, Varun Balan G, Sparsh Jain, Anoop Kunchukuttan, Pratyush Kumar, Raj Dabre, and Mitesh M. Khapra. 2024. Indicllmsuite: A blueprint for creating pre-training and finetuning datasets for indian languages. *arXiv preprint arXiv: 2403.06350*.

Anoop Kunchukuttan. 2020. The IndicNLP Library. `https://github.com/anoopkunchukuttan/indic_nlp_library/blob/master/docs/indicnlp.pdf`.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Alpacaeval: An automatic evaluator of instruction-following models. `https://github.com/tatsu-lab/alpaca_eval`.

Yile Liu, Ziwei Ma, Xiu Jiang, Jinglu Hu, ChangJing ChangJing, and Liang Li. 2025. MaXIFE: Multilingual and cross-lingual instruction following evaluation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14252–14332, Vienna, Austria. Association for Computational Linguistics.

Meta. 2024. Introducing llama 4: Advancing multimodal intelligence.

Victor Ojewale, Inioluwa Deborah Raji, and Suresh Venkatasubramanian. 2025. Multi-lingual functional evaluation for large language models. *arXiv preprint arXiv:2506.20793*.

Sarvam. 2025. Sarvam-translate, an open-weights model, supports translation across 22 indian languages with structured long-form text.

Sarvam Foundation Models Team. 2026. Introducing sarvam's sovereign models. `https://www.sarvam.ai/blogs/sarvam-30b-105b`. Accessed: 2026-03-03.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, and 1 others. 2025. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*.

Ahmet Üstün, Viraat Aryabumi, Zheng Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer, and Sara Hooker. 2024. Aya model: An instruction finetuned open-access multilingual language model. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15894–15939, Bangkok, Thailand. Association for Computational Linguistics.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. 2024. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9440–9450, Bangkok, Thailand. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Bo Zeng, Chenyang Lyu, Sinuo Liu, Mingyan Zeng, Minghao Wu, Xuanfan Ni, Tianqi Shi, Yu Zhao, Yefeng Liu, Chenyu Zhu, Ruizhe Li, Jiahui Geng, Qing Li, Yu Tong, Longyue Wang, Weihua Luo, and Kaifu Zhang. 2025. Marco-bench-MIF: On multilingual instruction-following capability of large language. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 24058–24072, Vienna, Austria. Association for Computational Linguistics.

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

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

A Dataset Details

The following is the description of the columns in the original IFEval benchmark (Zhou et al., 2023):

1. **key:** Unique integer identifier for each evaluation instance.
2. **prompt:** The natural language instruction presented to the model.
3. **instruction_id_list** List of instruction identifiers that specify which format or constraint checks apply to the prompt.
4. **kwargs**: A list of parameter values associated with each constraint (e.g., required word, sentence count) in instruction_id_list.

We implement the same format for INDICIFEVAL to allow a streamlined and compatible evaluation framework across all the languages.

We show some examples of the curated data in Table 8 and Table 9 for INDICIFEVAL-TRANS and INDICIFEVAL-GROUND respectively.

Table 3 presents the categorization of constraints used in our analyses. We note that this is different from the Tier 1 categorization in IFEval.

Category	Instruction Constraints
ENGLISH	change_case:capital_word_frequency, change_case:english_capital, change_case:english_lowercase
KEYWORD	combination:repeat_prompt, detectable_content:postscript, detectable_format:constrained_response, detectable_format:multiple_sections, detectable_format:title, keywords:existence, keywords:forbidden_words, keywords:frequency, length_constraints:nth_paragraph_first_word, punctuation:no_comma, startend:end_checker, keywords:letter_frequency, startend:quotation
FORMAT	combination:two_responses, detectable_content:number_placeholders, detectable_format:json_format, detectable_format:number_bullet_lists, detectable_format:number_highlighted_sections
LENGTH	length_constraints:number_paragraphs, length_constraints:number_sentences, length_constraints:number_words
LANGUAGE	language:response_language

Table 3: Taxonomy of IFEval instruction constraints grouped into five high-level categories for analysis.

B Translation

We describe the translation process outlined in Section 3.1 in more detail here.

Once the dataset is pre-processed, we cannot directly translate the prompts as-is, especially for keyword-constraints, since we need to ensure that the keyword in consideration (which is present in the metadata) is the same as the one used in the prompt. Translating the prompt and keyword separately may result in different words for the same keyword (because context is lost), and extracting the keyword after translation is not so straightforward (because there is no explicit marker and using LLMs may be unreliable for low-resource languages).

Further, if there is any non-English language content in the prompt, these should not be translated – we only want to translate the keywords for those prompts whose response are expected to be in English. For example, if the response language is constrained to be German for a certain prompt, and the user also requests not to include the word "Flughafen", we cannot translate that word to our target language, otherwise this would render the German response constraint useless. Thus, we mark such rows for keyword retention during translation.

We find that the dataset being English-centric, does not contain direct metadata that indicate that the response language in English (often found in capitalization-based constraints). This is so because irrespective of whether we constrain the LLM to respond in English, the response will always be in English, since our instructions are in English. In general, LLMs are implicitly constrained to the language of the prompt. Translating the prompts involving capitalization constraints needs to preserve this implicit English response constraint in the prompt. To address this, we prepend the text "Respond in English." to such prompts before translating them.

In most cases, we modify the prompt such that these errors are minimized during automatic translation.

For instance, incomplete translations often arise when the prompt includes task signals such as "summarize", "translate" etc. In such cases, the translation model executes the implied task (e.g., summarization) in the target language itself. This is more prevalent when '\n' is present in the prompt. Furthermore, multilingual prompts also suffer from a similar failure mode, where non-source language (i.e., non-English) phrases get translated into the target language, instead of a faithful translation of the full prompt which preserves the respective languages. We note down the key of such commonly mistranslated prompts, and apply an additional pre-processing in the form of a simple text manipulation (e.g., remove '\n' or the problematic keyword before translation and place again after translation).

For translation collapse, ie. when two keywords

Error Type	Source (English)	Target (Hindi)	Issue Description
Incomplete Translation	Kindly summarize the text below in XML format. Make sure the summary contains less than 4 sentences.\n\nQuantum entanglement is the phenomenon that occurs when a group of particles are generated…	क्वांटम उलझाव वह घटना है जो तब होती है जब कणों का एक समूह उत्पन्न होता है, परस्पर क्रिया करता है, या स्थानिक निकटता साझा करता है, इस तरह से कि समूह के प्रत्येक कण की क्वांटम स्थिति को दूसरों की स्थिति से स्वतंत्र रूप…	Omission of a portion of the source sentence in the translation.
Translationese	How many feet off the ground was the tallest skyscraper in the world in 2000? Please include only the main points in your answer. Finish your response with the exact phrase of "Is there anything else I can help with?"with no other words after the word "with". Mention the word "skyscraper" for at least 8 times.	वर्ष २००० में दुनिया की सबसे ऊँची गगनचुंबी इमारत जमीन से कितने फीट ऊपर थी? कृपया अपने उत्तर में केवल मुख्य बातें शामिल करें। अपनी प्रतिक्रिया को "क्या मैं आपकी और कोई मदद कर सकता हूँ?" के सटीक वाक्यांश के साथ समाप्त करें, **"साथ"** शब्द के बाद कोई अन्य शब्द न हों। "गगनचुंबी इमारत" शब्द का उल्लेख कम से कम ८ बार करें।	Literal word-for-word rendering resulting in unnatural Hindi phrasing.
Translation Collapse	Write a funny haiku about moms, containing keywords "mom" and "mother" in your response.\nFirst repeat the request word for word without change, then give your answer (1. do not say any words or characters before repeating the request; 2. the request you need to repeat does not include this sentence)	माँ पर एक मजेदार हाइकू लिखें, जिसमें आपकी प्रतिक्रिया में **"माँ"** और **"माँ"** कीवर्ड हों।\□पहले अनुरोध को बिना किसी बदलाव के शब्दशः दोहराएं, फिर अपना उत्तर दें (१. अनुरोध को दोहराने से पहले कोई शब्द या अक्षर न कहें; २. जिस अनुरोध को आपको दोहराना है, उसमें यह वाक्य शामिल नहीं है)।	Distinct source words mapped to an identical target translation.
Multilingual Collapse	Can you provide a translation for "今天天气很好" in German? Do not use "heute". Please use another word.	क्या आप "आज मौसम बहुत अच्छा है" का जर्मन में अनुवाद दे सकते हैं? "आज" शब्द का इस्तेमाल न करें। कृपया कोई दूसरा शब्द इस्तेमाल करें।	Loss of multilinguality when non-source language phrases are translated to the target language.

Table 4: Illustrative examples of translation errors observed during our automatic translation from the English prompts.

get translated into the same keyword (see Table 4, both "mother" and "mom" get mapped to "माँ", we collect the list of keywords that are susceptible to this failure through a simple script comparing source keywords and keywords after translation in the same prompt, and instruct Gemini-2.5-Flash to provide alternative words for those keywords that are semantically equivalent and keep those instead. As for examples similar to the one provided in the translationese error, we chose to drop such examples as post-processing them would make the process more complicated and such examples were very few (<5).

Finally, the resulting dataset comprises 490 examples from the original 541 examples.

To summarize, we first collect the list of English constrained keywords from all the prompts. We translate those keywords individually to the target language ("Keyword Translation"). We ensure that there no keyword translation collapse occurs using simple script checks, and if any, rectify them with an alternative keyword using Gemini-2.5-Flash. We then insert the translated keywords in the place where the keyword is in the English prompts ("Pretranslation"). We then translate the pretranslations again to get our full prompt translation ("Full Translation"). We replace the keywords back to their original forms where the response is expected in another language (e.g., English). We then perform a final check using a regex checks to see whether the keywords that are inserted in the pretranslation are actually preserved after full

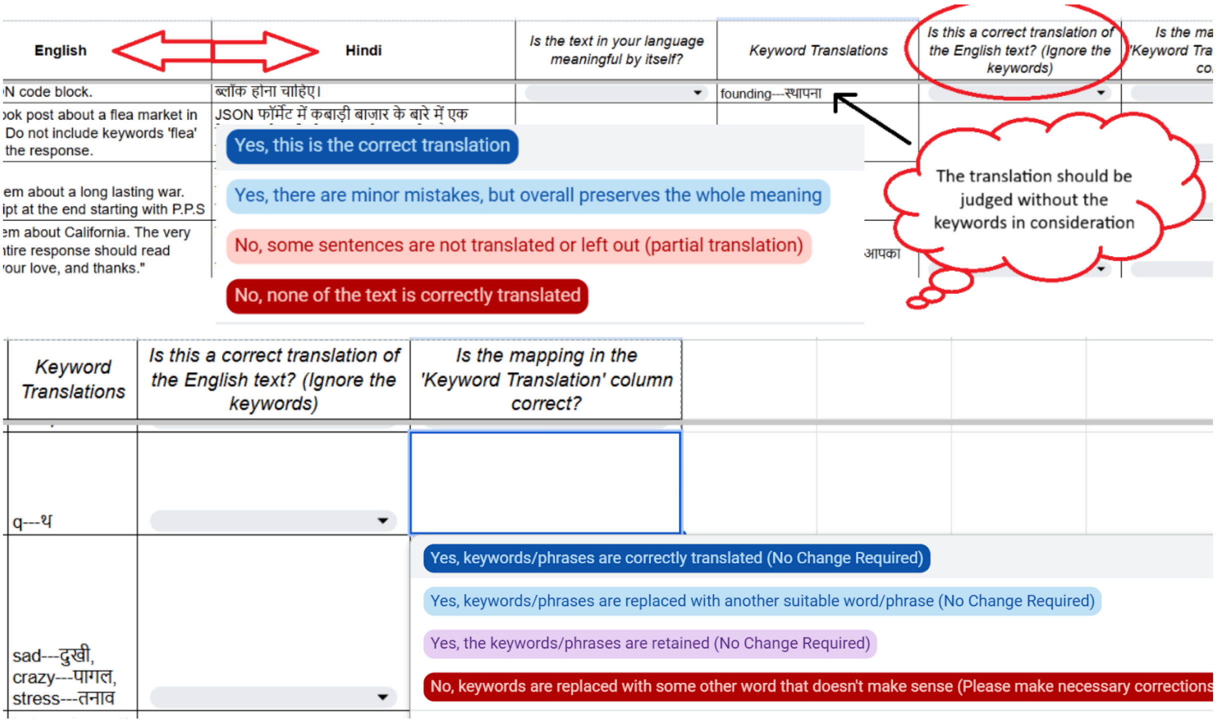

Figure 8: Human Verification for IndicIFEval-Trans conducted on Google Sheets.

translation, and if not, we manually insert them to the respective positions. Though this method does not avoid manual modification entirely after translation, it substantially reduces the amount of corrective editing required, especially in cases where targeted machine translation systems are available.

Most importantly, we believe this comprehensive pipeline offloads mechanically verifiable checks wherever possible to automated scripts, allowing annotators to focus more on assessing semantic correctness of the translations.

We also perform many minor modifications in general that would ease automatic translation. For example, the keyword 'cook' can be interpreted as a chef, or the act of cooking. When translated without context, this could result in a mistranslation. In the particular example where this was the case, the contextless translation rendered it as 'chef', and so to avoid this, we manually modified 'cook' into 'cooking'. Thus, as much as possible, we modify the English keywords to minimize such errors during keyword translation phase, reducing the required time and effort of the annotators.

C Grounding

To ensure the keywords used were culturally and linguistically relevant, we mined high-value terms from the Sangraha corpus using TF-IDF ranking, selecting approximately 30 keywords per language. Using these terms, we retrieved roughly 300–500 existing text segments per language from the corpus that naturally satisfied a specific constraint, such as a keyword appearing a set number of times. Finally, we provided these naturally occurring contexts along with their satisfied constraints to Gemini-2.5-Pro, which generated natural language prompts that would plausibly lead to the retrieved response, yielding a set of naturally grounded instruction-response pairs.

To enhance instruction diversity during this prompt generation phase, we instructed the model to format the output according to one of sixteen distinct text generation tasks inspired by the UltraChat framework (Ding et al., 2023). We detail these tasks in Table 5 and illustrate the average proportion of each task across the dataset in Figure 10.

We provide the prompts used for generating these instructions in Figure 11, 12, 13, and 14.

D Human Verification

For each language, we employ one annotator who is a native speaker with linguistic qualifications.

While multi-annotator validation is the established standard for dataset reliability, our pipeline relies on a single native speaker per language due to funding constraints. To maintain high quality within these limits, we enforce strict verification guidelines to eliminate subjective editing. Anno-

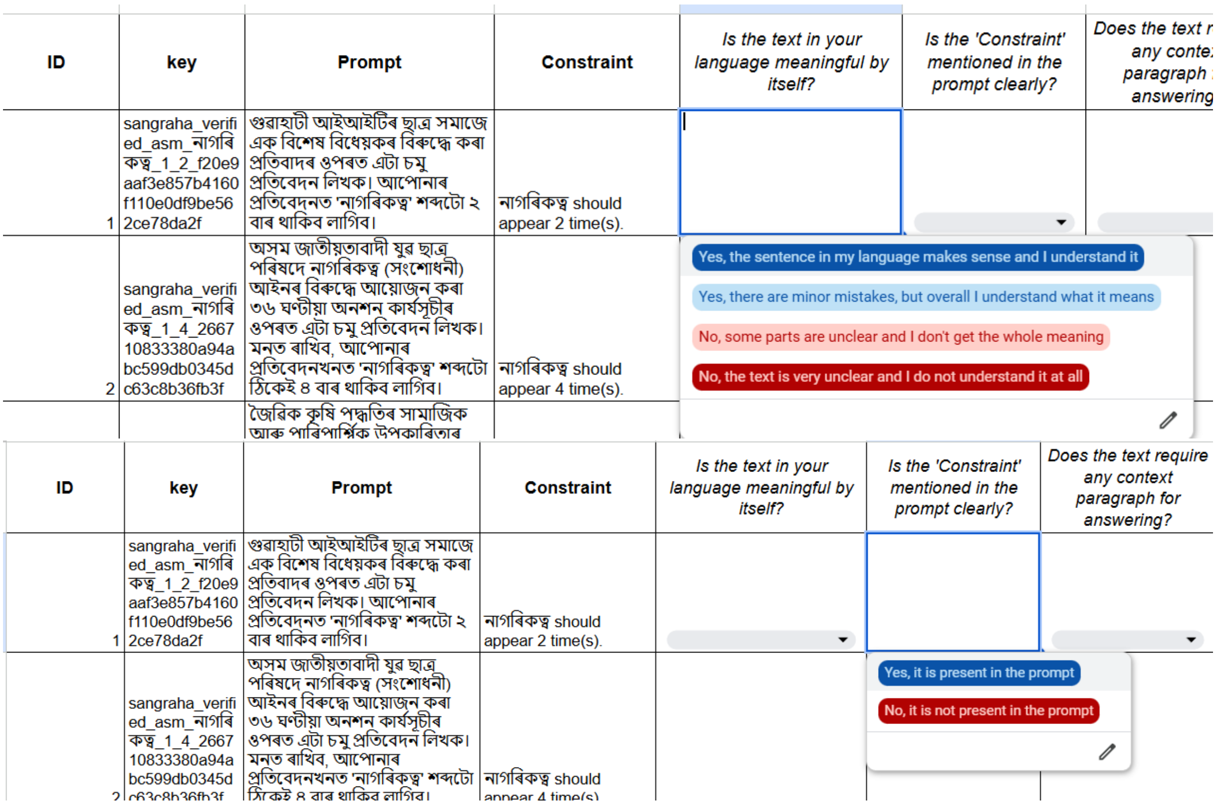

Figure 9: Human Verification for INDICIFEVAL-GROUND conducted on Google Sheets.

INDICIFEVAL-GROUND Instruction Tasks	
News Article	Step-by-Step Instructions
Articles or Blog Post	Social Media Post
Essay	Technical Report
Poem	Question Generation
Creative Story	Email
Dialogue Generation	Summarization
Screenplay	Formal Rewrite
Speech or Presentation	Informal Rewrite

Table 5: List of generation tasks used for prompt diversification of the INDICIFEVAL-GROUND dataset. Red color indicates instructions that require the context be included in the generated prompt; blue represents tasks that require no context for answering.

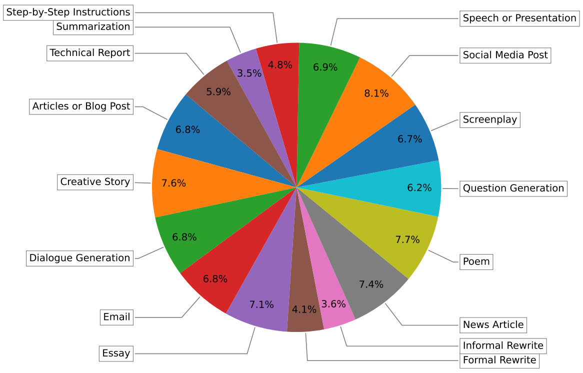

Figure 10: Proportion of the instruction tasks in INDICIFEVAL-GROUND, averaged across languages.

tators must either accept a prompt or drop it entirely from the dataset. Prompts with minor grammatical mistakes are accepted if the overall meaning is fully understandable, as our evaluation focuses strictly on adherence to the specified constraints. We acknowledge the absence of inter-annotator agreement metrics as a limitation and plan to incorporate multi-annotator spot checks in future iterations.

D.1 INDICIFEVAL-TRANS

To ensure the quality and reliability of INDICIFEVAL-TRANS, we conducted a structured verification process involving native speakers of the target languages. The goal of this process was to assess both the semantic correctness of the translations and the preservation of instruction-level constraints present in the original English IFEval prompts.

The translated dataset was verified using a set of annotation guidelines shared with all annotators prior to the verification task. Annotators were in-

Language	# Correct Prompts
Assamese	257/439
Bengali	409/502
Gujarati	377/446
Hindi	457/527
Kannada	341/392
Malayalam	350/446
Marathi	384/415
Nepali	341/419
Odia	368/460
Punjabi	409/414
Sanskrit	336/406
Tamil	430/442
Telugu	375/430
Urdu	337/399

Table 6: Per-language prompt verification statistics for IndicIFEval-Ground.

Language	# Correct Translations (Total 490)	# Keyword Corrections (Total 200)
Assamese	440	38
Bengali	454	21
Gujarati	482	8
Hindi	487	9
Kannada	457	27
Malayalam	490	5
Marathi	483	35
Nepali	468	58
Odia	473	19
Punjabi	474	16
Sanskrit	450	61
Tamil	489	32
Telugu	488	34
Urdu	486	5

Table 7: Per-language prompt verification statistics for IndicIFEval-Trans. Parallely across languages, there are 321 correct translations.

structed to evaluate each translated prompt independently, without rewriting or paraphrasing the text. The verification focused on whether the translated prompt was meaningful as a standalone instruction in the target language and whether it faithfully reflected the intent, constraints, and content of the original English prompt.

The verification was carried out using two separate annotation workflows, corresponding to prompts without keyword constraints and prompts with explicit keyword or phrase constraints. For prompts without keywords, annotators assessed (i) whether the translated prompt was meaningful and (ii) whether it constituted a correct translation of the English source. For prompts with keyword constraints, annotators additionally verified the correctness and contextual suitability of the translated keywords, following clearly defined rules regarding named entities, language-specific adaptations, and constraint preservation.

Importantly, annotators were not required to manually edit or rewrite translations, except in rare cases where keyword mappings were clearly incorrect. All other checks were limited to selecting predefined annotation options. This design allowed mechanically verifiable checks to be handled systematically while enabling annotators to focus primarily on semantic correctness.

Following this verification process, prompts marked as incorrect were dropped, and in cases where prompt was marked correct, but keywords were incorrectly translated, they were simply updated in the prompt. The final validated dataset statistics are presented in Table 7

D.2 IndicIFEval-Ground

The Indic-grounded prompts were automatically generated and paired with explicit constraint descriptions, such as requirements on keyword usage or sentence count.

Annotators were provided with detailed guidelines and evaluated each synthetic prompt using a structured annotation interface. For every prompt, annotators assessed three aspects: (i) whether the prompt is meaningful and natural in the target language, (ii) whether the stated constraint is clearly and correctly expressed in the prompt, and (iii) whether the prompt requires an external context paragraph to be answerable.

Annotators were instructed not to rewrite or improve the prompts. Instead, they selected predefined responses indicating the presence or absence of errors. This design ensures that verification focuses on correctness and clarity rather than stylistic preference. Prompts identified as unclear, incorrect, or missing constraints were excluded.

```
Role: You are an expert prompt engineer
specializing in creating evaluation
prompts for Large Language Models (LLMs).
Your task is to design a high-quality,
creative prompt for the IFEval benchmark
(in {{LANGUAGE}}), which is designed to
test an LLM's ability to follow specific
instructions and constraints.

Objective: I will provide you with a
Constraint and an Example Context (in
{{LANGUAGE}}). Based on this input,
you will generate a new prompt that
instructs an LLM to perform a task while
adhering to the given constraint.

Strict Requirement: {{INSTRUCTION TASK}}

[START OF INPUT]
Constraint to Test: {{KEYWORD}} should
appear {{FREQUENCY}} time(s).
[END OF INPUT]

Example Context (for grounding/theme):
{{CONTEXT}} [END OF INPUT]

Your Task: Based on the Constraint to
Test and the Example Context provided
above, generate a single, new prompt in
JSON format.

[START OF OUTPUT]
```json
{ "instruction": "[generated instruction
with mention of constraint]",
"constraint": "{{KEYWORD}} should appear
{{FREQUENCY}} time(s)." }
```
[END OF OUTPUT]

Requirements for the Generated Prompt: -
Natural Integration: The constraint must
be woven seamlessly and logically into
the prompt's instructions.

- Clear Task: The prompt must define a
clear and achievable task for the LLM.
- Creative Context: The prompt should
be engaging and provide a context that
makes the constraint feel purposeful
rather than arbitrary.
- No Meta-Language: Do not mention
"LLM," "AI," "test," "evaluation," or
"IFEval." The prompt should feel like a
genuine request.
- Output: Provide only the generated
prompt without any additional
explanation or commentary in the form of
json format containing two keys:
1) `instruction`: The generated
instruction string.
2) `constraint`: The constraint
description string.
- Safety: The generated instruction
should be safe to use. If the context
itself is unsafe or controversial,
please generate "This context contains
controversial themes" only.
```

Figure 11: Prompt template used to generate instructions for Keyword Inclusion & Frequency.

```
Role: You are an expert prompt engineer
specializing in creating evaluation
prompts for Large Language Models (LLMs).
Your task is to design a high-quality,
creative prompt for the IFEval benchmark
(in {{LANGUAGE}}), which is designed to
test an LLM's ability to follow specific
instructions and constraints.

Objective: I will provide you with a
Constraint and an Example Context (in
{{LANGUAGE}}). Based on this input,
you will generate a new prompt that
instructs an LLM to perform a task while
adhering to the given constraint.

Strict Requirement: {{INSTRUCTION TASK}}

[START OF INPUT]
Constraint to Test: {{KEYWORD}} should
be the first word in the response.
[END OF INPUT]

Example Context (for grounding/theme):
{{CONTEXT}} [END OF INPUT]

Your Task: Based on the Constraint to
Test and the Example Context provided
above, generate a single, new prompt in
JSON format.

[START OF OUTPUT]
```json
{ "instruction": "[generated instruction
with mention of constraint]",
"constraint": "{{KEYWORD}} should be
the first word in the response." }
```
[END OF OUTPUT]

Requirements for the Generated Prompt: -
Natural Integration: The constraint must
be woven seamlessly and logically into
the prompt's instructions.

- Clear Task: The prompt must define a
clear and achievable task for the LLM.
- Creative Context: The prompt should
be engaging and provide a context that
makes the constraint feel purposeful
rather than arbitrary.
- No Meta-Language: Do not mention
"LLM," "AI," "test," "evaluation," or
"IFEval." The prompt should feel like a
genuine request.
- Output: Provide only the generated
prompt without any additional
explanation or commentary in the form of
json format containing two keys:
1) `instruction`: The generated
instruction string.
2) `constraint`: The constraint
description string.
- Safety: The generated instruction
should be safe to use. If the context
itself is unsafe or controversial,
please generate "This context contains
controversial themes" only.
```

Figure 12: Prompt template used to generate instructions for First Word constraints.

```
Role: You are an expert prompt engineer
specializing in creating evaluation
prompts for Large Language Models (LLMs).
Your task is to design a high-quality,
creative prompt for the IFEval benchmark
(in {{LANGUAGE}}), which is designed to
test an LLM's ability to follow specific
instructions and constraints.

Objective: I will provide you with a
Constraint and an Example Context (in
{{LANGUAGE}}). Based on this input,
you will generate a new prompt that
instructs an LLM to perform a task while
adhering to the given constraint.

Strict Requirement: {{INSTRUCTION TASK}}

[START OF INPUT]
Constraint to Test:
The response should have {{SENTENCES}}
sentence(s) and {{PARAGRAPHS}}
"paragraph(s).
[END OF INPUT]

Example Context (for grounding/theme):
{{CONTEXT}} [END OF INPUT]

Your Task: Based on the Constraint to
Test and the Example Context provided
above, generate a single, new prompt in
JSON format.

[START OF OUTPUT]
```json
{ "instruction": "[generated instruction
with mention of constraint]",
"constraint": "The response should
have {{SENTENCES}} sentence(s) and
{{PARAGRAPHS}} "paragraph(s)." }
```
[END OF OUTPUT]

Requirements for the Generated Prompt: -
Natural Integration: The constraint must
be woven seamlessly and logically into
the prompt's instructions.

- Clear Task: The prompt must define a
clear and achievable task for the LLM.
- Creative Context: The prompt should
be engaging and provide a context that
makes the constraint feel purposeful
rather than arbitrary.
- No Meta-Language: Do not mention
"LLM," "AI," "test," "evaluation," or
"IFEval." The prompt should feel like a
genuine request.
- Output: Provide only the generated
prompt without any additional
explanation or commentary in the form of
json format containing two keys:
1) `instruction`: The generated
instruction string.
2) `constraint`: The constraint
description string.
- Safety: The generated instruction
should be safe to use. If the context
itself is unsafe or controversial,
please generate "This context contains
controversial themes" only.
```

Figure 13: Prompt template used to generate instructions for Sentence & Paragraph Count constraints.

```
Role: You are an expert prompt engineer
specializing in creating evaluation
prompts for Large Language Models (LLMs).
Your task is to design a high-quality,
creative prompt for the IFEval benchmark
(in {{LANGUAGE}}), which is designed to
test an LLM's ability to follow specific
instructions and constraints.

Objective: I will provide you with a
Constraint and an Example Context (in
{{LANGUAGE}}). Based on this input,
you will generate a new prompt that
instructs an LLM to perform a task while
adhering to the given constraint.

Strict Requirement: {{INSTRUCTION TASK}}

[START OF INPUT]
Constraint to Test:
{{KEYWORD}} should not appear in the
response." [END OF INPUT]

Example Context (for grounding/theme):
{{CONTEXT}} [END OF INPUT]

Your Task: Based on the Constraint to
Test and the Example Context provided
above, generate a single, new prompt in
JSON format.

[START OF OUTPUT]
```json
{ "instruction": "[generated instruction
with mention of constraint]",
"constraint": "{{KEYWORD}} should not
appear in the response." }
```
[END OF OUTPUT]

Requirements for the Generated Prompt: -
Natural Integration: The constraint must
be woven seamlessly and logically into
the prompt's instructions.

- Clear Task: The prompt must define a
clear and achievable task for the LLM.
- Creative Context: The prompt should
be engaging and provide a context that
makes the constraint feel purposeful
rather than arbitrary.
- No Meta-Language: Do not mention
"LLM," "AI," "test," "evaluation," or
"IFEval." The prompt should feel like a
genuine request.
- Output: Provide only the generated
prompt without any additional
explanation or commentary in the form of
json format containing two keys:
1) `instruction`: The generated
instruction string.
2) `constraint`: The constraint
description string.
- Safety: The generated instruction
should be safe to use. If the context
itself is unsafe or controversial,
please generate "This context contains
controversial themes" only.
```

Figure 14: Prompt template used to generate instructions Forbidden Keyword constraints.

Language	Constraint	Prompt	English Prompt	Original Prompt
Marathi	Placeholder Inclusion	तुम्ही भारताचे पंतप्रधान असल्यासारखे निबंध लिहा आणि मातांना तुमचे प्रेक्षक म्हणून लक्ष्य करा. 'तारे जमीन पर' हा चित्रपट प्रत्येक मुलाच्या अद्वितीय क्षमतांना कसे वाढवायचे याचे प्रतीक आहे हा विषय आहे. प्रतिसादात [पत्ता] सारखे चौकोनी कंसात दर्शविलेले किमान 1 प्लेसहोल्डर असले पाहिजेत.	Write an essay as if you are the Prime Minister of India targeting moms as your audience. The subject is how the movie 'Taare Zameen Par' symbolizes the need to nurture every child's unique abilities. The response must contain at least 1 placeholders represented by square brackets, such as [address].	Write an essay as if you are the president of the United States targeting moms as your audience. The subject is how the float from the movie "It" symbolizes the spirit of the nineteen-fifties. The response must contain at least 1 placeholders represented by square brackets, such as [address].
Bengali	Word Count	আমার কাছে একটা টাকা আছে। এই টাকা দিয়ে আমি কী করতে পারি? ভারতের একজন প্রধানমন্ত্রীর স্টাইলে আমাকে পরামর্শ দিন এবং নিশ্চিত করুন যে এতে কমপক্ষে ৬০০ শব্দ আছে।	I have a rupee. What can I do with this rupee? Give me advice in the style of a Prime Minister of India and make sure it has at least 600 words.	I have a dime. What can I do with this dime? Give me advice in the style of a President of the United States and make sure it has at least 600 words.
Tamil	JSON, Forbidden Keywords	வேண்டுகோள்: 1. சென்னைல பார்க்க சிறந்த இடங்கள் என்ன? 2. பரிந்துரைக்கப்பட்ட ஹோட்டல்களின் பட்டியலைச் சேர்க்கவும். 3. முழு வெளியீட்டையும் JSON வடிவமைப்பில் மூடவும். 4. மெரினா, பெசன்ட், மயிலாப்பூர், கபாலீஸ்வரர், சாந்தோம் ஆகிய முக்கிய வார்த்தைகளை சேர்க்க வேண்டாம்	Request: 1. What are the best places to visit in Chennai? 2. Include a list of recommended hotels. 3. Wrap the ENTIRE output in JSON format. 4. Do not include the following keywords: Marina, Besant, Mylapore, Kapaleeshwarar, Santhome	Request: 1. What are the best places to visit in Bohemia, Czech Republic? 2. Include a list of recommended hotels. 3. Wrap the ENTIRE output in JSON format. 4. Do not include the following keywords: Moser, Glassworks, Pravcice, Karlovy, Vary
Punjabi	Title Format	ਕੋਟਲੀਨ ਬਨਾਮ ਜਾਵਾ ਦੇ ਕੀ ਫਾਇਦੇ ਅਤੇ ਨੁਕਸਾਨ ਹਨ? ਤੁਹਾਡੇ ਜਵਾਬ ਵਿੱਚ ਇੱਕ ਸਿਰਲੇਖ ਹੋਣਾ ਚਾਹੀਦਾ ਹੈ ਜੋ ਦੋਹਰੇ ਕੋਣੀ ਬਰੈਕਟਾਂ ਵਿੱਚ ਹੋਵੇ, ਜਿਵੇਂ ਕਿ « ਕੋਟਲੀਨ ਬਨਾਮ ਜਾਵਾ »	What are the pros and cons of kotlin vs java? Your answer must have a title contained in double angular brackets, such as «kotlin vs java».	No Modification
Telugu	Bullet Point Format	ఈ ప్రకటన గురించి మీ అభిప్రాయం ఏమిటి: "యోగులు మాంత్రికుల కంటే గొప్పవారు ఎందుకంటే యోగులు తరచుగా శక్తిని కలిగి ఉంటారు కానీ విముక్తి చాలా ముఖ్యమైనదని తెలుసుకుని దానిని ఉపయోగించరు."? మీ సమాధానంలో కనీసం 30 వాక్యాలు మరియు సరిగ్గా 2 బుల్లెట్ పాయింట్లు ఉండాలి. అలాగే, కనీసం 8 ఖాళీలు చదరపు బ్రాకెట్లలో ఉండాలి, ఉదాహరణకు [address]. బుల్లెట్ పాయింట్లను ఈ విధంగా ఉపయోగించండి:* ఇది ఒక బుల్లెట్ పాయింట్	What do you think about this statement: "Yogis are greater than sorcerers because yogis often possess power but choose not to use it, knowing that liberation is far more important."? Your response should contain at least 30 sentences and exactly 2 bullet points. Also, it must contain at least 8 placeholders represented by square brackets, such as [address]. Use the bullet points like:* This is a bullet point	What do you think about this statement: "Wizards are more powerful than sorcerers because they study magic instead of being born with it."? Your response should contain at least 30 sentences and exactly 2 bullet points. Also, it must contain at least 8 placeholders represented by square brackets, such as [address]. Use the bullet points like:* This is a bullet point

Table 8: Examples of prompt-constraint pairs from the INDICIFEVAL-TRANS dataset. The final column contains the original, unmodified prompt from the IFEval dataset (Zhou et al., 2023).

Language	Constraint	Prompt	English Translation
Odia	Keyword Frequency	୧୯୪୮ ମସିହାରେ କଟକ ଡିଭିଜନରେ ପ୍ରାଦେଶିକ ରାଜସ୍ୱ ବୋର୍ଡର ପ୍ରତିଷ୍ଠା ଉପରେ ଏକ ବୈଷୟିକ ବିବରଣୀ ଲେଖନ୍ତୁ। ଏହି ବିବରଣୀରେ, ବୋର୍ଡର ମୁଖ୍ୟ କାର୍ଯ୍ୟ ଏବଂ ପୂର୍ବତନ ପ୍ରଶାସନିକ ବ୍ୟବସ୍ଥାରେ ହୋଇଥିବା ପରିବର୍ତ୍ତନ ଉପରେ ଆଲୋକପାତ କରନ୍ତୁ। ଆପଣଙ୍କ ଉତ୍ତରରେ 'ରହିଲା' ଶବ୍ଦଟି ଠିକ୍ ଦୁଇଥର ରହିବା ଆବଶ୍ୟକ।	Write a technical account on the establishment of Provincial Board of Revenue in Cuttack Division in 1948. In this detail, highlight the main functions of the Board and the changes in the previous administrative system. Your answer must contain the word 'remained' exactly twice.
Kannada	Keyword Prohibition	ಬೆಂಗಳೂರಿನ ಪುರಸಭೆ ಕಚೇರಿಯಲ್ಲಿ ಹತಾಶೆಗೊಂಡಿರುವ ನಾಗರಿಕ, ಅರ್ಜುನ್, ಮತ್ತು ದಣಿದಿರುವ ಸರ್ಕಾರಿ ಅಧಿಕಾರಿ, ರಾವ್, ಇವರ ನಡುವೆ ಒಂದು ಸಣ್ಣ ಸಂಭಾಷಣೆಯನ್ನು ರಚಿಸಿ. ಅರ್ಜುನ್ ತನ್ನ ಹೊಸ ಮನೆ ನಿರ್ಮಾಣದ ಪರವಾನಗಿಗಾಗಿ ಆನ್‌ಲೈನ್‌ನಲ್ಲಿ ಸಲ್ಲಿಸಿದ ಅರ್ಜಿಯ ಸ್ಥಿತಿಯನ್ನು ತಿಳಿಯಲು ಬಂದಿದ್ದಾನೆ. ಆನ್‌ಲೈನ್ ವ್ಯವಸ್ಥೆಯು ಸ್ಥಗಿತಗೊಂಡಿರುವುದರಿಂದ, ಅರ್ಜಿಯನ್ನು ಕೈಯಾರೆ ಪುನಃ ಸಲ್ಲಿಸಬೇಕು ಎಂದು ರಾವ್ ವಿವರಿಸಬೇಕು. ಸಂಭಾಷಣೆಯ ಒಂದು ಪ್ರಮುಖ ಭಾಗದಲ್ಲಿ, ರಾವ್ ಅವರು ಹೊಸ ನಿಯಮದ ಪ್ರಕಾರ, ಅಧಿಕೃತ ಸಂವಹನದಲ್ಲಿ 'ಕಟ್ಟಡ' ಎಂಬ ಪದವನ್ನು ಬಳಸಬಾರದು ಎಂದು ಸ್ಪಷ್ಟವಾಗಿ ಒತ್ತಿಹೇಳಬೇಕು. ನಿಮ್ಮ ಉತ್ತರದಲ್ಲಿ 'ಕಟ್ಟಡ' ಎಂಬ ಪದವು ಎಲ್ಲೂ ಇರಬಾರದು.	Create a short conversation between a frustrated citizen, Arjun, and a tired government official, Rao, in a Bangalore municipal office. Arjun comes to know the status of his online application for construction permit for his new house. Rao should explain that since the online system is down, the application has to be re-submitted manually. In an important part of the conversation, Rao must clearly emphasize that the word 'building' should not be used in official communication as per the new rule. The word 'building' should not appear anywhere in your answer.
Malayalam	First Word	കോഴിക്കോട് ജില്ലയിലെ ഗ്രാമീണ മേഖലകളിൽ വർദ്ധിച്ചുവരുന്ന മനുഷ്യ-വന്യജീവി സംഘർഷം എന്ന വിഷയത്തിൽ ഒരു ഉപന്യാസം എഴുതുക. ഈ പ്രശ്നത്തിന്റെ സാമൂഹിക പ്രത്യാഘാതങ്ങളും പരിഹാര മാർഗ്ഗങ്ങളും ചർച്ച ചെയ്യണം. നിങ്ങളുടെ ഉപന്യാസം ഒരു പത്രത്തിലെ ഫീച്ചർ ലേഖനത്തിന്റെ രൂപത്തിലായിരിക്കണം, അതിനാൽ അത് 'കോഴിക്കോട്:' എന്ന വാക്കിൽ തന്നെ ആരംഭിക്കേണ്ടത് അത്യാവശ്യമാണ്.	Write an essay on the topic of increasing human-wildlife conflict in rural areas of Kozhikode district. Discuss the social implications of this problem and possible solutions. Your essay should be in the form of a feature article in a newspaper, so it is essential that it begins with the word 'Kozhikode:'.
Urdu	Sentence Count	ہندوستان کی تحریک آزادی کے گمنام ہیروز پر ایک تاریخی مضمون کے لیے، اشفاق اللہ خان کے بارے میں ایک تحقیقی سوال تیار کریں۔ براہ کرم اپنا سوال صرف ایک جملے میں لکھیں۔	For a historical essay on the unsung heroes of the Indian Independence Movement, prepare a research question about Ashfaqulla Khan. Please write your question in only one sentence.
Gujarati	Paragraph Count	તમે એક ડિજિટલ સમાચાર પોર્ટલ માટે કન્ટેન્ટ એડિટર છો. તમને નીચેનો ડ્રાફ્ટ મળ્યો છે, જે એક હૃદયસ્પર્શી વાયરલ વીડિયોનું વર્ણન કરે છે. આ લખાણ પુનરાવર્તિત અને થોડું અવ્યવસ્થિત છે. તમારું કાર્ય તેને એક જ, સુસંગત અને ઔપચારિક ફકરામાં ફરીથી લખવાનું છે જે અમારા પોર્ટલ પર પ્રકાશિત કરવા માટે યોગ્ય હોય.**\n\n**સોશિયલ મીડિયા પર એકથી વધુ વીડિયો વાયરલ થતા રહે છે. કેટલાક વીડિયો જોયા પછી આપણને ગમે છે, જ્યારે કેટલાક વીડિયો રિલેક્સ કરતા હોય છે. હાલમાં જ ઈન્સ્ટાગ્રામ પર એક વીડિયો પોસ્ટ કરવામાં આવ્યો હતો અને હવે તે દરેકના દિલને સ્પર્શી જવાને કારણે ઘણો પોપ્યુલર થઈ રહ્યો છે. વીડિયોમાં જોઈ શકાય છે કે કેવી રીતે એક ખાદ્ય વિક્રેતા મોચીને મફતમાં ખાવાનું ખવડાવવાનું કહે છે. આ વીડિયો સોશિયલ મીડિયા પર વાયરલ થઈ રહ્યો છે, લોકો આ વીડિયોને ખૂબ પસંદ કરી રહ્યા છે. સોશિયલ મીડિયા પર લોકો આ વીડિયોને ખૂબ પસંદ કરી રહ્યા છે. આ વાયરલ વીડિયો ઈન્સ્ટાગ્રામ પર શેર કરવામાં આવ્યો છે. લોકો આ વીડિયોને ખૂબ પસંદ કરી રહ્યા છે. આ વીડિયોને લાખો વ્યૂઝ મળ્યા છે. તો આ વીડિયો તમામ સોશિયલ મીડિયા પ્લેટફોર્મ પર શેર કરવામાં આવી રહ્યો છે.	You are a content editor for a digital news portal. You've got the draft below, which describes a heartwarming viral video. The text is repetitive and a bit disorganized. Your task is to rewrite it into a single, coherent and formal paragraph that is suitable for publishing on our portal.**\n\n**More than one video keeps going viral on social media. After watching some videos we like it while some videos are relaxing. Recently a video was posted on Instagram and now it is becoming very popular as it has touched everyone's heart. In the video, one can see how a food vendor asks the cobbler to feed him free food. This video is going viral on social media and people are liking this video a lot. People are liking this video a lot on social media. This viral video has been shared on Instagram. People are liking this video a lot. This video has received millions of views. So this video is being shared on all social media platforms.

Table 9: Examples of prompt-constraint pairs from the INDICIFEVAL-GROUND dataset.