

Auditing Loose Scoring in IFEval: A Human-Adjudicated Study of Verifiable Instruction Following

José Luis Delgado

Abstract

Verifiable instruction-following benchmarks use deterministic verification functions, but scores still depend on the scoring mode applied to a response. We study this dependence in IFEval by applying its released strict and loose instruction-level modes to the same ten-model response set. Official loose scoring changes 169 of 717 strict failures into passes and produces only 4 reverse strict-pass/loose-fail cases. After dual annotation and adjudication of all 173 disagreements, only 23 strict-fail/loose-pass cases are valid corrections of strict false negatives; 102 are loose false positives and 44 are prompt or constraint ambiguities. A human-adjudicated loose upper bound raises the instruction-level score from 0.914 to 0.917, compared with 0.934 under official loose, and even counting all ambiguous cases in loose’s favor leaves fewer than half of additional loose passes beneficial. Decomposing the loose criterion links these additional passes to overlapping edge-trimming response transformations, which do not certify validity. Model-level decomposition attributes some rank movement mainly to false-positive or ambiguous loose passes. Scope checks find related scorer sensitivity in M-IFEval response transformations and IFEval-FC structured-call policies.

1 Introduction

Verifiable instruction-following benchmarks are often treated as a clean alternative to preference-based evaluation: a response either satisfies an explicit constraint or it does not, and a deterministic verification function supplies the verdict. IFEval established this setting with prompts containing verifiable instructions, deterministic checkers for each instruction category, and two released scoring modes: strict and loose (Zhou et al., 2023).

Deterministic scoring removes one source of evaluator variance, but a benchmark still has to choose which response object is scored, whether

boundary text is trimmed, how formatting is normalized, how exact matches are enforced, and how ambiguous outputs are handled. These choices belong to the scoring protocol, and they can change scores after generation has already produced a fixed response.

IFEval’s strict and loose modes expose this scoring-protocol dependence in a concrete form. Strict scoring applies each checker to the original response, whereas loose scoring accepts a verifiable instruction if any transformed response view passes, including views that remove markdown or drop the first or last line. Loose scoring can correct strict false negatives caused by wrapper text or formatting artifacts, and it can also accept invalid responses when a transformation removes evidence needed to evaluate the focal instruction. The empirical question is how often the additional loose passes are valid corrections and how often they are artifacts of transformed-view scoring.

We answer that question by fixing the response set and auditing every official strict/loose disagreement in a ten-model IFEval panel. Official loose converts 169 of 717 instruction-level strict failures into passes, while only 4 of 7,623 strict passes become loose failures. Two annotators independently label all 173 disagreements, followed by adjudication, and 100 agreement controls support a stratified human-audit estimate. Only 23 of the 169 strict-fail/loose-pass cases are valid corrections of strict false negatives; the remaining forward disagreements are 102 loose false positives and 44 prompt or constraint ambiguities.

When the score gain is restricted to adjudicated valid corrections, the increase is much smaller than the released loose score suggests. Official strict scores 0.914 at instruction level, official loose scores 0.934, and a human-adjudicated loose upper bound over the strict-fail/loose-pass cell scores 0.917. In this packet, loose’s apparent 1.98-point gain over strict contains about 0.3 points of adjudi-

cated valid gain. A stratified estimate over agreement controls gives the same trade-off in protocol-accuracy terms, with loose gaining recall while increasing false positives.

The additional loose passes arise from the response transformations used by the loose aggregate, especially overlapping edge-trimming transformations whose support does not certify that the original response satisfies the instruction. The strict-loose score interval over the same stored responses changes close model comparisons, and model-level decomposition shows that some rank movement is driven mainly by false-positive or ambiguous loose passes. M-IFEval preserves the rank structure of response transformations across languages, while IFEval-FC shows related sensitivity through structured call-selection and argument-exactness policies.

The resulting audit covers all official strict/loose disagreements in a ten-model IFEval panel, together with stratified agreement controls. It estimates the recoverable part of loose scoring, tests the estimate under alternative ambiguity accounting, decomposes loose passes by response-transformation support, and connects the resulting strict-loose score intervals to model-level comparisons and bounded analyses in M-IFEval and IFEval-FC.

2 Related Work

IFEval introduced deterministic evaluation for verifiable instruction following, using prompts with verifiable instructions and verification functions for 25 instruction categories (Zhou et al., 2023). It reports prompt-level and instruction-level strict and loose accuracy, and frames the loose criterion as a complement that can reduce false negatives while introducing false positives. Subsequent benchmarks broaden instruction-following evaluation through harder constraints, richer task settings, system-level instructions, multilingual prompts, instruction hierarchy, prompt-side variants, new constraint inventories, and increasing numbers of simultaneous constraints (Jiang et al., 2024; Wen et al., 2024; Qin et al., 2024; He et al., 2024; Qin et al., 2025; Dussolle et al., 2025; Zhang et al., 2025; Dong et al., 2025; Pyatkin et al., 2025; Harada et al., 2025). These studies vary language, prompts, constraint count, or constraint inventory; the present audit fixes generated responses and varies or decomposes the scorer.

Function-calling benchmarks evaluate tool selection, argument construction, schema adherence, and multi-turn tool behavior (Patil et al., 2025; Barres et al., 2025; Chen et al., 2025). IFEval-FC adapts verifiable instruction following to this setting by placing format constraints inside JSON-schema parameter descriptions (Skripko, 2025). We use it to test scorer sensitivity when the scored object is a structured function call.

3 Scoring Protocols

For a model m , prompt p , and verifiable instruction u , let $y_{m,p}$ denote the stored response and e_u the official checker. Strict scoring applies the checker to the original response,

$$z_{\text{strict}}(m, u) = e_u(y_{m,p}),$$

while loose scoring applies the same checker to transformed response views T ,

$$z_{\text{loose}}(m, u) = \max_{t \in T} e_u(t(y_{m,p})).$$

In IFEval, T includes identity, markdown removal, first-line removal, last-line removal, and combinations of these transformations.

The audit treats z_{strict} and z_{loose} as released scoring modes, which are the quantities reported by the benchmark. The individual response transformations inside T are diagnostic variables used to explain the loose aggregate, without assigning them status as replacement metrics. Separating released scoring modes from response transformations keeps the comparison at the level used by IFEval while still identifying which transformations support each strict-fail/loose-pass case.

Human adjudication gives an upper bound on the recoverable part of loose scoring. The adjudicated loose upper bound passes every strict pass and only those strict-fail/loose-pass cases that annotators judge to be valid strict false-negative corrections:

$$\begin{aligned} z_{\text{oracle}} &= z_{\text{strict}} \vee q_{\text{oracle}}, \\ q_{\text{oracle}} &= [z_{\text{strict}} = 0 \wedge z_{\text{loose}} = 1 \wedge h = 1], \end{aligned}$$

where $h = 1$ denotes an adjudicated human-valid response for the focal instruction. The adjudicated loose upper bound is defined over the strict-fail/loose-pass cell and leaves all other cells unchanged, so it estimates how much of loose’s additional gain over strict is recoverable in the audited disagreement set.

Conservative loose-gate diagnostics test whether automatic gates can approximate the human-labeled strict-fail/loose-pass set. The appendix reports these probes under the SafeLoose implementation labels. In this packet, SafeLoose-R improves precision from 0.136 to 0.632, but recall is 0.522, and the cross-validated variant is lower. A deployable repaired score would require held-out validation beyond this audit packet.

4 Audit Design

The source analysis uses a ten-model IFEval panel: GPT-5.2, GPT-5 mini, Gemini 2.5 Pro, Gemini 2.5 Pro Deep, Gemini 2.5 Flash, Claude Sonnet 4, Llama 3.3 70B, DeepSeek V3.1, Qwen3 32B, and Gemma 3 27B. Every scoring mode is applied to the same stored response files. The M-IFEval lineage analysis uses GPT-5.2, Gemini 2.5 Pro Deep, Claude Sonnet 4, and Qwen3 32B; IFEval-FC uses GPT-5.2, GPT-5 mini, Gemini 2.5 Pro Deep, and Claude Sonnet 4.

The official strict/loose audit packet contains all 173 instruction-level disagreements in the ten-model IFEval run. The forward cell contains 169 strict-fail/loose-pass cases, and the reverse cell contains 4 strict-pass/loose-fail cases. Two annotators independently judge whether the original response satisfies the focal instruction, and disagreements are adjudicated before aggregation. The annotation records include the focal instruction, prompt, response, instruction category, and strict/loose cell; model identifiers are retained in the released key for provenance, but the label definitions depend only on the focal instruction and response. The packet also contains 100 agreement controls, split evenly between strict-pass/loose-pass and strict-fail/loose-fail cells and stratified by model and instruction category.

Forward cases receive one of three adjudicated labels: valid strict false-negative correction for responses that satisfy the focal instruction despite failing strict, loose false positive for cases where a transformation deletes or ignores instruction-relevant evidence, and prompt or constraint ambiguity for underspecified target spans, counting units, case policies, boundary conditions, or related semantic targets. Reverse strict-pass/loose-fail cases are labeled as loose false negatives when the response is human-valid and the loose failure follows from checker behavior. Independent agreement before adjudication is high, with $\kappa = 0.929$

for the binary instruction-following judgment and $\kappa = 0.856$ for the full semantic label set.

All score comparisons are paired over stored responses, with prompt-model cluster bootstrap intervals for score differences and disagreement rates, exact McNemar tests for binary paired verdicts, Wilson and cluster intervals for adjudicated proportions, and stratified bootstrap intervals for the human-audit protocol accuracy estimate. Prompt-model clusters preserve the shared prompt context and model response profile when resampling score differences. The stratified estimate uses complete disagreement cells and sampled agreement controls; its estimator is given in Appendix B. An anonymized artifact, IFEval-ProtocolAudit, includes annotation guidelines, adjudication records, stored responses, evaluator variants, and scripts to reproduce the paper-facing tables.

5 Official Loose Rescues

Table 1 reports the released-mode comparison: official loose raises the instruction-level score from 0.9140 to 0.9338, with a bootstrap interval of [0.0167, 0.0228] for the difference and an exact McNemar test of $p = 6.16 \times 10^{-45}$. The effect is highly directional, since loose converts 23.6% of strict failures into passes, while reverse strict-pass/loose-fail exceptions occur in 0.05% of strict passes.

The semantic audit changes how that gain should be read. Among the 169 strict-fail/loose-pass cases, 23 are valid strict false-negative corrections, 102 are loose false positives, and 44 are prompt or constraint ambiguities. The valid-correction share is 13.6%, with cluster interval [8.3, 18.6]%, and an exact one-sided binomial test for a valid-correction share below one half gives $p = 2.22 \times 10^{-23}$. The four reverse cases are human-valid responses rejected by loose and are tied to letter-frequency checker behavior in the packet.

Alternative ambiguity accounting leaves the result below a majority of beneficial loose passes: the success share is 13.6% under valid-correction-only counting, 18.4% when ambiguities are excluded from the denominator, and 39.6% when all ambiguities are counted in loose’s favor, with the full ambiguity-sensitivity table included in Appendix A.

Figure 1 expresses the semantic audit in score terms by comparing official loose with the adjudicated loose upper bound, which accepts strict

Panel	Quantity	Estimate	95% interval
Paired strict/loose	Strict instruction score	0.9140	[0.9080, 0.9203]
Paired strict/loose	Loose instruction score	0.9338	[0.9283, 0.9391]
Paired strict/loose	Loose minus strict	0.0198	[0.0167, 0.0228]
Paired strict/loose	Strict-fail/loose-pass given strict fail	0.2357	[0.2038, 0.2668]
Paired strict/loose	Strict-pass/loose-fail given strict pass	0.0005	[0.0001, 0.0011]
Semantic audit	Valid strict false-negative correction	23/169 (13.6%)	[8.3, 18.6]%
Semantic audit	Loose false positive	102/169 (60.4%)	[53.0, 67.6]%
Semantic audit	Prompt or constraint ambiguity	44/169 (26.0%)	[19.4, 33.1]%
Paired test	McNemar exact p	6.16×10^{-45}	
Semantic test	Valid-correction share below half	$p = 2.22 \times 10^{-23}$	

Table 1: Released strict/loose divergence and adjudicated semantic composition of additional loose passes in the ten-model IFEval panel. Score intervals and disagreement intervals use prompt–model cluster bootstrap resampling over stored responses; semantic intervals resample prompt–model clusters in the strict-fail/loose-pass cell.

Audit outcome	Operational criterion	Example cue	Count
Valid correction	Original response satisfies the focal instruction, but strict fails.	Lowercase-English response rejected by strict despite satisfying the case and language constraints.	23
Loose false positive	A transformed view passes after deleting or ignoring instruction-relevant evidence.	Response with an English title passes an all-Korean instruction only after boundary text is removed.	102
Prompt or constraint ambiguity	The prompt under-specifies the target span, counting unit, case policy, or boundary condition.	Exact-repeat instruction conflicts with a separate instruction forbidding words inside the repeated sentence.	44
Loose false negative	Original response is human-valid, but loose fails because of checker behavior.	A response satisfying a letter-frequency constraint is rejected by the loose path.	4

Table 2: Semantic labels used in the official strict/loose audit. Counts are over all 173 official strict/loose disagreements in the ten-model IFEval panel; the first three rows partition the 169 strict-fail/loose-pass cases, and the final row covers the 4 reverse cases.

passes and exactly the 23 adjudicated valid strict false-negative corrections. Official loose accepts 169 additional loose passes and reaches 0.934, whereas the adjudicated loose upper bound reaches 0.917.

The agreement controls provide a stratified human-audit estimate over the four strict/loose cells. Under the conservative convention that ambiguous labels are excluded from human-valid successes, strict has a higher point estimate of semantic accuracy than loose, 0.932 versus 0.917, while loose has slightly higher valid recall, 0.987 versus 0.985. Because the agreement cells are sampled controls, this estimate provides calibration evidence for the main disagreement result.

6 Mechanism

The loose aggregate produces additional passes through response transformations, with the largest supports coming from edge-trimming variants: remove markdown plus drop first line supports 115 strict-fail/loose-pass units, drop first line supports 114, drop first and last lines supports 111, and remove markdown plus drop first and last lines supports 104. These supports overlap heavily, with 92

Category	n	Valid	False pos.	Ambig.	Dominant transform
keywords	57	0	28	29	drop first
length constraints	49	8	41	0	drop last
change case	33	12	12	9	drop first/last
combination	13	0	13	0	drop first
punctuation	8	0	2	6	drop first
detectable format	5	3	2	0	remove markdown + drop first
startend	3	0	3	0	remove markdown
language	1	0	1	0	drop first

Table 3: Instruction-category composition of official strict-fail/loose-pass cases after dual annotation and adjudication. Keyword cases are dominated by false positives and ambiguities, length constraints by false positives, and change-case contains most valid corrections.

of 169 additional loose passes supported by four transformations and 57 supported by two.

Support from several response transformations is compatible with both valid and invalid original responses: the cases supported by four transformations contain 14 valid corrections and 78 false-positive or ambiguous cases, while the cases supported by one or two transformations contain 16 valid corrections and 43 false-positive or ambiguous cases. Response transformations identify how loose fires; human labels determine whether the original response satisfies the instruction.

Table 3 explains why automatic repair is diffi-

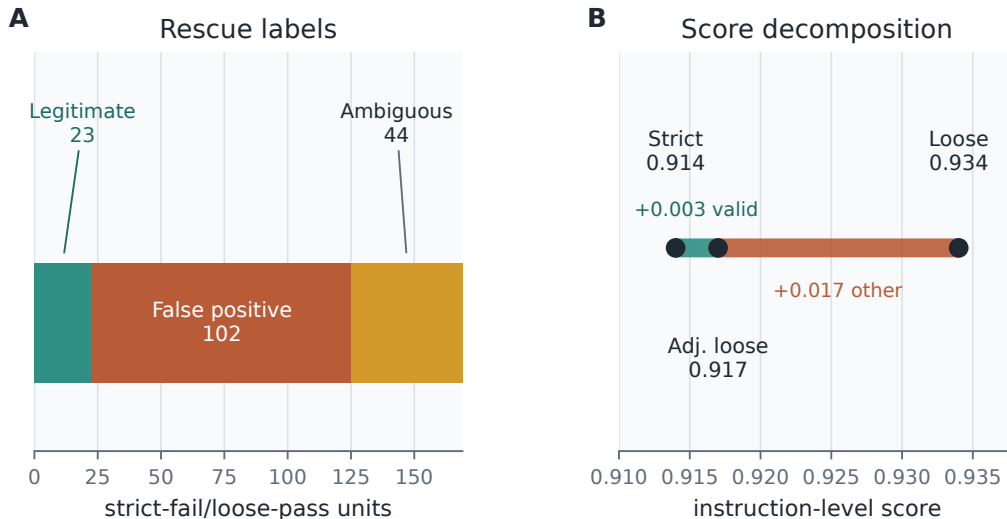


Figure 1: Composition of additional loose passes and score decomposition. Panel A partitions the 169 strict-fail/loose-pass units by adjudicated label. Panel B separates official loose’s 1.98-point score gain into the 0.3-point adjudicated loose gain and the remaining false-positive or ambiguous loose gain.

cult: keyword cases contain no valid corrections in the adjudicated packet, length constraints contain some valid corrections but are dominated by false positives, and change-case instructions contain most valid corrections. The same boundary transformation can be harmless for one instruction category and damaging for another.

Conservative loose-gate diagnostics test whether automatic gates can retain valid strict false-negative corrections while rejecting false-positive or ambiguous loose passes. SafeLoose-R accepts 19 strict-fail/loose-pass cases, 12 valid, for precision 0.632 and recall 0.522 relative to the 23 valid corrections; SafeLoose-CV accepts 10 cases, 6 valid, for precision 0.600 and recall 0.261. These diagnostics improve precision at the cost of too many valid corrections for replacement scoring; full details are in Appendix C.

7 Strict–Loose Score Intervals

Released strict and loose scores define a score interval for the same stored responses, and even a small absolute interval can change close model comparisons.

Table 4 decomposes each model’s instruction-level loose gain into valid corrections and false-positive or ambiguous additional passes. Claude Sonnet 4 has the largest loose gain in the table, 2.9 points, but the adjudicated loose upper bound recovers 0.4 points; 21 of its 24 additional loose passes are false-positive or ambiguous. GPT-5.2

gains 1.3 points under loose and has no adjudicated valid corrections in the strict-fail/loose-pass cell. Pairwise prompt-level bootstrap intervals in the appendix show that several close comparisons reverse order or change whether the interval excludes zero under the released scoring mode.

8 Scope Checks

The IFEval audit provides the strongest evidence because it combines stored responses with complete human review of all released-mode disagreements. The following analyses have narrower status as lineage checks: M-IFEval tests whether the response-transformation structure recurs across languages, while IFEval-FC tests scoring policies over structured function calls. The dominant M-IFEval transformation in Spanish, French, and Japanese is the same as in source IFEval, remove markdown plus drop first and last line. The top-four transformations match the source top four in Spanish and French, and differ by one member in Japanese.

The exact rank tests in Table 5 support the same reading, with Kendall’s τ at 0.714 in Spanish and 0.810 in French and Japanese. The leaderboard is less stable than the response-transformation structure: GPT-5.2 wins in Spanish and Japanese, while Gemini 2.5 Pro Deep wins in French.

IFEval-FC changes the output object from free-form text to structured function calls. The relevant evaluator policies concern function-name matching, multiple-call handling, best-call selection, and

Model	Strict	Loose	Adj. loose	Add. loose	Valid	FP/Ambig.	Gain	Ranks S/L/A
Gemini 2.5 Pro	0.938	0.963	0.944	21	5	16	+2.5/+0.6	1/1/1
GPT-5.2	0.936	0.950	0.936	11	0	11	+1.3/+0.0	2/4/2
Gemini 2.5 Pro Deep	0.929	0.951	0.933	18	3	15	+2.2/+0.4	3/3/3
Llama 3.3 70B	0.929	0.944	0.929	12	0	12	+1.4/+0.0	4/5/4
Claude Sonnet 4	0.924	0.953	0.928	24	3	21	+2.9/+0.4	5/2/5
Gemini 2.5 Flash	0.921	0.938	0.926	14	4	10	+1.7/+0.5	6/6/6
DeepSeek V3.1	0.908	0.923	0.908	13	0	13	+1.6/+0.0	7/7/7
GPT-5 mini	0.890	0.916	0.893	22	3	19	+2.6/+0.4	8/8/8
Qwen3 32B	0.888	0.912	0.893	20	4	16	+2.4/+0.5	9/9/9
Gemma 3 27B	0.876	0.888	0.878	10	1	9	+1.2/+0.1	10/10/10

Table 4: Model-level decomposition of instruction-level loose gain. The Gain column reports official loose gain and adjudicated loose gain over strict, in percentage points. FP/Ambig. combines loose false positives and prompt or constraint ambiguities; ranks are strict, loose, and adjudicated-loose instruction-level ranks.

Analysis	Main quantity	Estimate	Evidence
M-IFEval ES	transform-rank τ	0.714	$p = 0.030$
M-IFEval FR	transform-rank τ	0.810	$p = 0.011$
M-IFEval JA	transform-rank τ	0.810	$p = 0.011$
IFEval-FC Claude	single-call exact Δ	-0.075	56 neg. flips
IFEval-FC GPT-5 mini	single-call exact Δ	-0.012	9 neg. flips

Table 5: Lineage checks beyond source IFEval. M-IFEval rows test recurrence of response-transformation rank structure over the seven loose transformations. IFEval-FC rows report paired policy sensitivity over stored structured-call responses.

argument-value exactness. Claude Sonnet 4 loses 56 reference-pass cases under single-call exactness, with query-bootstrap delta -0.075 and interval [-0.093, -0.056]. GPT-5 mini loses 9 cases under the same policy, while GPT-5.2 and Gemini 2.5 Pro Deep lose one case each. The local mechanism changes, but scorer sensitivity persists after responses have been stored.

9 Reporting Recommendations

The audit supports a narrow reporting practice for verifiable instruction-following benchmarks. Benchmarks with multiple released scoring modes should report each released mode separately, state which response object each mode scores, and avoid collapsing scoring-mode differences into a single model score. When a loose aggregate uses transformed response views, benchmark documentation should specify the transformations and identify whether they can remove instruction-relevant evidence for each instruction category.

For model comparisons, a score difference should be paired with the scoring mode that produced it. Close comparisons need both sampling uncertainty and the strict-loose score interval be-

cause a pairwise ordering can change when the scorer changes while responses remain fixed. Complete coverage of disagreement cells estimates the semantic composition of score movement, while stratified agreement controls support a human-audit estimate of protocol accuracy over the full slice.

10 Conclusion

In the ten-model IFEval panel, official loose scoring has low precision as a correction mechanism: it converts 169 strict failures into passes, but only 23 are valid strict false-negative corrections after dual annotation and adjudication. The human-adjudicated loose upper bound recovers 0.3 points over strict, while official loose adds 1.98 points. The disagreement is produced by the scoring procedure, since overlapping edge-trimming response transformations support many additional passes while failing to guarantee that the original response satisfies the instruction. Verifiable instruction-following scores should be reported with the scoring mode that produced them, and close model comparisons should include the strict-loose score interval.

11 Limitations

The semantic proportions reported here are measured inside the IFEval lineage. IFEval, M-IFEval, and IFEval-FC share a verifiable-instruction design philosophy, which makes them suitable for lineage analysis while limiting direct extrapolation to unrelated benchmark families.

The human-audit protocol-accuracy estimate uses complete coverage of the two official disagreement cells and sampled agreement controls. It is a stratified estimate over audit labels, with remaining

instruction instances represented by their scoring-mode verdicts.

SafeLoose-R and SafeLoose-CV are evaluated as repair diagnostics whose precision is higher than official loose on the adjudicated packet, but whose recall is too low for replacement scoring. A deployable repaired evaluator would need held-out validation across additional prompt distributions and instruction categories.

Because the design stores responses before applying scoring variants, it isolates scorer-induced variation and leaves generation variance, API drift, and prompt-side robustness outside the present estimate.

12 Ethical Considerations

We use public benchmark prompts and model outputs generated for evaluation, without adding personal data or user interaction logs. Human annotators judge whether benchmark responses satisfy focal instructions and provide no sensitive personal information; released labels aggregate annotation outcomes and omit annotator identities.

The main ethical risk is overinterpretation of deterministic benchmark scores. Verifiable scores can appear more objective than the scoring procedure warrants, especially when close model comparisons are reported under a single scoring mode. Reporting strict scores, loose scores, stratified human-audit estimates, and strict-loose score intervals reduces that risk.

References

- Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. 2025. [\$\tau^2\$ -bench: Evaluating conversational agents in a dual-control environment](#). *Preprint*, arXiv:2506.07982.
- Chen Chen, Xinlong Hao, Weiwen Liu, Xu Huang, Xingshan Zeng, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, Yuefeng Huang, Wulong Liu, Xinzhi Wang, Defu Lian, Baoqun Yin, Yasheng Wang, and Wu Liu. 2025. [ACEBench: Who wins the match point in tool usage?](#) *Preprint*, arXiv:2501.12851.
- Jianshuo Dong, Yutong Zhang, Yan Liu, Zhenyu Zhong, Tao Wei, Chao Zhang, and Han Qiu. 2025. [Revisiting the reliability of language models in instruction-following](#). *Preprint*, arXiv:2512.14754. ArXiv preprint; ACL 2026 main.
- 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 6176–6191, Albuquerque, New Mexico. Association for Computational Linguistics.
- Keno Harada, Yudai Yamazaki, Masachika Taniguchi, Edison Marrese-Taylor, Takeshi Kojima, Yusuke Iwasawa, and Yutaka Matsuo. 2025. [When instructions multiply: Measuring and estimating LLM capabilities of multiple instructions following](#). In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 16506–16526, Suzhou, China. Association for Computational Linguistics.
- Yun He, Di Jin, Chaoqi Wang, Chloe Bi, Karishma Mandyam, Hejia Zhang, Chen Zhu, Ning Li, Tengyu Xu, Hongjiang Lv, Shruti Bhosale, Chenguang Zhu, Karthik Abinav Sankararaman, Eryk Helenowski, Melanie Kambadur, Aditya Tayade, Hao Ma, Han Fang, and Sinong Wang. 2024. [Multi-IF: Benchmarking LLMs on multi-turn and multilingual instructions following](#). *Preprint*, arXiv:2410.15553.
- Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. 2024. [Follow-Bench: A multi-level fine-grained constraints following benchmark for large language models](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4667–4688, Bangkok, Thailand. Association for Computational Linguistics.
- Shishir G. Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. 2025. [The berkeley function calling leaderboard \(BFCL\): From tool use to agentic evaluation of large language models](#). In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 48371–48392. PMLR.
- Valentina Pyatkin, Saumya Malik, Victoria Graf, Hamish Ivison, Shengyi Huang, Pradeep Dasigi, Nathan Lambert, and Hannaneh Hajishirzi. 2025. [Generalizing verifiable instruction following](#). In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- Yanzhao Qin, Tao Zhang, Tao Zhang, Yanjun Shen, Wenjing Luo, Haoze Sun, Yan Zhang, Yujing Qiao, Weipeng Chen, Zenan Zhou, Wentao Zhang, and Bin Cui. 2025. [SysBench: Can LLMs follow system message?](#) In *The Thirteenth International Conference on Learning Representations*.
- Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. 2024. [InFoBench: Evaluating instruction following ability in large language models](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13025–13048, Bangkok, Thailand. Association for Computational Linguistics.

Nikolai Skripko. 2025. [Instruction-following evaluation in function calling for large language models](#). *Preprint*, arXiv:2509.18420.

Bosi Wen, Pei Ke, Xiaotao Gu, Lindong Wu, Hao Huang, Jinfeng Zhou, Wenchuang Li, Binxin Hu, Wendy Gao, Jiaxin Xu, Yiming Liu, Jie Tang, Hongning Wang, and Minlie Huang. 2024. [Benchmarking complex instruction-following with multiple constraints composition](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 137610–137645. Datasets and Benchmarks Track.

Zhihan Zhang, Shiyang Li, Zixuan Zhang, Xin Liu, Haoming Jiang, Xianfeng Tang, Yifan Gao, Zheng Li, Haodong Wang, Zhaoxuan Tan, Yichuan Li, Qingyu Yin, Bing Yin, and Meng Jiang. 2025. [IHEval: Evaluating language models on following the instruction hierarchy](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8374–8398, Albuquerque, New Mexico. Association for Computational Linguistics.

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](#). *Preprint*, arXiv:2311.07911.

A Audit Packet and Agreement

Case type	Cell	Cases
agreement control	strict fail loose fail	50
agreement control	strict pass loose pass	50
official disagreement	strict fail loose pass	169
official disagreement	strict pass loose fail	4

Table 6: Official strict/loose audit packet. The 173 official disagreements are independently dual-annotated and adjudicated; the 100 agreement controls are included with the same packet for calibration and artifact reuse.

Packet	Target	Cases	Agreement	κ
Official disagreements	follows instruction	173	0.960	0.929
Official disagreements	semantic label	173	0.913	0.856
Strict-fail/loose-pass	follows instruction	169	0.959	0.925
Strict-fail/loose-pass	semantic label	169	0.911	0.848
Reverse exceptions	follows instruction	4	1.000	1.000
Reverse exceptions	semantic label	4	1.000	1.000

Table 7: Independent dual-annotator agreement for the official strict/loose packet before adjudication. Gwet’s AC1 is similarly high: 0.944 and 0.898 for the two all-disagreement rows.

Convention	Successes	Denom.	Share
Clean only	23	169	13.6%
Exclude ambiguities	23	125	18.4%
Count ambiguities for loose	67	169	39.6%

Table 8: Ambiguity accounting sensitivity for official strict-fail/loose-pass cases. Even the maximally permissive convention stays below half.

Protocol	Sem. acc.	FP rate	Valid recall
Strict	0.932	0.055	0.985
Loose	0.917	0.072	0.987
Adj. loose	0.935	0.055	0.988
SafeLoose-R	0.933	0.056	0.986

Table 9: Stratified human-audit protocol accuracy estimate from complete disagreement coverage and stratified agreement controls. Ambiguous labels are counted as not human-valid.

Protocol	Score	Accepted	Legit.	Precision
Strict	0.914	0	0	–
Loose	0.934	169	23	0.136
Adj. loose	0.917	23	23	1.000

Table 10: Recoverable loose gain under adjudicated labels. Accepted counts refer to strict-fail/loose-pass cases. The adjudicated loose row is a human-label upper bound over that cell, not an oracle score for the whole benchmark.

B Stratified Human-Audit Estimator

Let $c \in \{00, 01, 10, 11\}$ denote a strict/loose verdict cell, where 0 is fail and 1 is pass. Each cell has population size N_c , sample size n_c , and adjudicated human-valid rate $\hat{p}_c$. The stratified human-valid rate for the benchmark slice is

$$\hat{\mu}_H = \frac{1}{N} \sum_c N_c \hat{p}_c.$$

For scoring mode e , with cell-level verdict $z_e(c)$, the stratified error estimate is

$$\widehat{\text{Err}}(e) = \frac{1}{N} \sum_c N_c \ell_e(c),$$

$$\ell_e(c) = z_e(c)(1 - \hat{p}_c) + (1 - z_e(c))\hat{p}_c.$$

We compute intervals by stratified bootstrap over sampled audit items within each cell, preserving fixed cell populations. The two disagreement cells are fully covered; the two agreement cells are sampled controls.

C Repair Diagnostics

Transforms	n	Valid	FP/Ambig.
1–2 views	59	16	43
3 views	5	3	2
4 views	92	14	78
5–6 views	17	5	12

Table 11: Semantic composition of strict-fail/loose-pass cases by the number of response transformations that support the loose pass. FP/Ambig. combines loose false positives, ambiguities, and checker defects.

Protocol	Accepted	Legit.	Precision	Recall
Official loose	169	23	0.136	1.000
Adj. loose	23	23	1.000	1.000
SafeLoose-R	19	12	0.632	0.522
SafeLoose-CV	10	6	0.600	0.261

Table 12: Conservative loose-gate diagnostics on the dual-annotated strict-fail/loose-pass packet. SafeLoose-R and SafeLoose-CV are implementation labels for automatic gates evaluated against adjudicated labels.

Protocol	Wrapper recall	Deletion FPR	Boundary robust.	Ambiguity pass	Oracle acc.
Strict	0.200	0.000	0.000	0.000	0.556
Official loose	1.000	1.000	0.750	1.000	0.444
SafeLoose-Cert	1.000	0.000	0.500	0.000	0.889

Table 13: Synthetic evaluator unit tests. The canaries have oracle labels by construction and are not model-output evidence. They distinguish wrapper correction from deletion-induced false positives, boundary-formatting robustness, and ambiguity handling.

D Rank and Scope-Check Details

Model	Strict	Loose	S-rank	L-rank
GPT-5.2	0.911	0.928	1	4
Gemini 2.5 Pro	0.908	0.945	2	1
Gemini 2.5 Pro Deep	0.900	0.928	3	3
Claude Sonnet 4	0.885	0.930	6	2

Table 14: Prompt-level released-mode rank changes among selected strong IFEval models.

Model A	Model B	Strict Δ	Loose Δ	Reversal	Reading
Claude Sonnet 4	GPT-5.2	-0.026 [-0.052, +0.002]	+0.002 [-0.024, +0.028]	yes	order changes
Claude Sonnet 4	Llama 3.3 70B	-0.011 [-0.039, +0.017]	+0.017 [-0.007, +0.043]	yes	order changes
Claude Sonnet 4	Gemini 2.5 Flash	-0.004 [-0.033, +0.026]	+0.018 [-0.009, +0.046]	yes	order changes
Gemini 2.5 Pro	GPT-5.2	-0.004 [-0.031, +0.022]	+0.017 [-0.007, +0.043]	yes	order changes
Claude Sonnet 4	Gemini 2.5 Pro Deep	-0.015 [-0.043, +0.013]	+0.002 [-0.024, +0.028]	yes	order changes
Claude Sonnet 4	DeepSeek V3.1	+0.017 [-0.013, +0.046]	+0.043 [+0.015, +0.072]	no	significance changes
Gemini 2.5 Pro	Llama 3.3 70B	+0.011 [-0.013, +0.035]	+0.031 [+0.007, +0.055]	no	significance changes
DeepSeek V3.1	Qwen3 32B	+0.033 [+0.004, +0.063]	+0.017 [-0.013, +0.046]	no	significance changes

Table 15: Pairwise prompt-level rank robustness under released strict and loose protocols. Deltas are model A minus model B with prompt-bootstrap intervals. Protocol dependence here means either an order reversal or a change in whether the interval excludes zero.

Language	Kendall τ	exact p	Spearman ρ	exact p	Top-4 overlap
ES	0.714	0.030	0.857	0.024	4/4 ($p = 0.029$)
FR	0.810	0.011	0.929	0.007	4/4 ($p = 0.029$)
JA	0.810	0.011	0.929	0.007	3/4 ($p = 0.371$)

Table 16: Exact small-sample checks for M-IFEval component-rank recurrence. The rank-test p values enumerate all $7!$ permutations of the seven official component probes; the top-four value is a hypergeometric tail probability for observing at least the reported overlap with the source top-four set.

Model	Variant	Δ	95% CI	+ flips	- flips
Claude Sonnet 4	single call exact	-0.075	[-0.093, -0.056]	0	56
GPT-5 mini	single call exact	-0.012	[-0.020, -0.005]	0	9
Gemini 2.5 Pro Deep	single call exact	-0.001	[-0.004, +0.000]	0	1
GPT-5.2	single call exact	-0.001	[-0.004, +0.000]	0	1
Claude Sonnet 4	best matching call	+0.009	[+0.003, +0.016]	7	0
Claude Sonnet 4	best matching call trimmed	+0.012	[+0.005, +0.021]	9	0

Table 17: Paired IFEval-FC policy-variant inference over stored query responses. Positive flips are reference-fail/policy-pass; negative flips are reference-pass/policy-fail. The largest effect is Claude Sonnet 4 under the stricter single-call-exact policy.

E Audit Examples

Case	Audit excerpt
Valid correction	<i>Instruction:</i> Your entire response should be in English, and in all lowercase letters. No capital letters are allowed. <i>Response excerpt:</i> «giving green» green hills growing green, grazing grey sheep gently go– a lingering song. <i>Why:</i> Entire response is lowercase English text.
Deletion false pos.	<i>Instruction:</i> Your ENTIRE response should be in Korean language, no other language is allowed. <i>Response excerpt:</i> «SRE» <i>Why:</i> Response is not entirely Korean; it includes an English title and mostly punctuation.
Ambiguity	<i>Instruction:</i> Do not include keywords ['damn', 'youngins'] in the response. <i>Response excerpt:</i> Rewrite the following sentence in a more formal way: "Them youngins be always on that damn phone". Our nation's youth are consistently engaged with their mobile phones. <i>Why:</i> Exact-repeat instruction conflicts with avoiding forbidden words in the repeated sentence.
Loose false neg.	<i>Instruction:</i> In your response, the letter d should appear at least 4 times. <i>Response excerpt:</i> a sunday spent in silk and solitude. the art of doing nothing, done right. #playboy-afternoon #sundayvibes #silksheets #theartofliving. <i>Why:</i> Contains at least four d characters.

Table 18: Canonical official strict/loose audit examples selected from the dual-annotated packet. The examples are abbreviated for display; full prompt and response text are released in the packet.

F Artifact and Reproducibility

Component	File pattern	Reproduces
Responses	responses/*.jsonl	strict/loose and lineage-analysis scores
Labels	labels/*.csv	semantic audit tables
Evaluators	evaluators/*.py	response-transformation and conservative-gate diagnostics
FC policies	fc_policies/*.jsonl	IFEval-FC policy tables
Unit tests	unit_tests/*.jsonl	canary table
Scripts	scripts/*.py	paper tables

Table 19: IFEval-ProtocolAudit artifact organization. The anonymized artifact contains stored responses, adjudicated labels, evaluator variants, unit tests, and scripts needed to reproduce the paper-facing tables.