neoAI-InstructBench: 実践的シナリオに基づく日本語複合指示追従ベンチマーク

川本 稔己 [1] 板井 孝樹 [1,2] 大槻 真輝 [1,3]
[1] 株式会社 neoAI [2] 東京都立大学 [3] 東京大学
{t.kawamoto, k.itai, m.otsuki}@neoai.jp

概要

実運用環境における大規模言語モデルには, 形式, 文体, 内容など複数の制約を同時に満たす制御能力が求められる. しかし, 既存のベンチマークは定型的な指示や, 単一カテゴリー中心の設計であり, 実世界の多様で複合的な指示を十分に反映していない. そこで本研究では, 5 つのカテゴリにまたがり, 指示文が重複しない複合指示ベンチマーク neoAI-InstructBench を構築した. 評価の結果, GPT-5.2 におけるタスク完遂率は 67%に留まり, その難易度が明らかとなった. さらに分析により, 指示間干渉の傾向と, 表記・形式制約がボトルネックとなり得ることを示した. 本ベンチマークと評価用コードは公開する[1)].

1 はじめに

大規模言語モデル（LLM）の社会実装が進むにつれ, 実利用シーンにおけるユーザーの指示は複雑化している. タスク 1 つをとっても, 箇条書き形式, 専門用語の回避, 比喩の挿入といったように, 多岐にわたる制約が同時に課されることが一般的である. 特に日本語では, 敬語体系やカタカナ等の表記. 文末表現といった言語固有の制約が存在するため, 複合的な制約を満たす能力を実運用に近い形で評価することが重要である.

しかし, 既存の指示追従能力評価ベンチマークは, 実運用で現れる複合指示を十分に反映できていない. IfEval [1] のような代表的ベンチマークは, 文字数制限や禁止語, 指定フォーマットといった客観的に判定可能な制約（Closed-ended [2]）を中心に設計されており, 判定に文脈理解や解釈を要する非決定論的な制約（Open-ended）を評価できない. また, 評価に用いる制約の種類や表現が限られている場合, モデルがそのパターンに過学習し, ベンチマーク上の遵守が実運用における頑健な指示追従能力を必ずしも保証しないことが報告されている [3]. さらに, 多くの既存設定 [2, 4] は単一の観点からの指示で構成されるため, 実利用で頻出する異種制約の同時要求や複数制約が互いに及ぼす干渉を検証できていない.

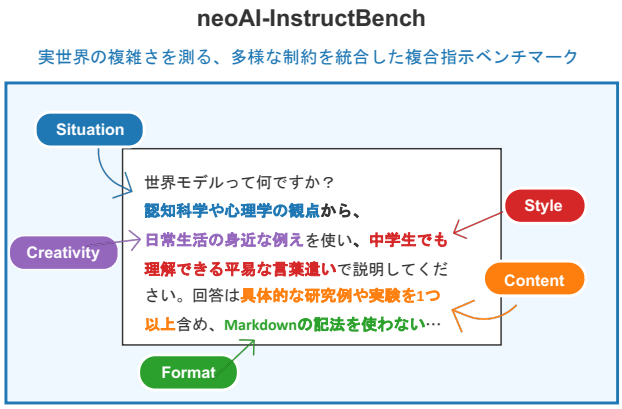

図 1 neoAI-InstructBench の概要

本研究は, 実運用シナリオに基づいた汎用的な日本語複合指示を体系的に評価し, モデルの指示追従能力の実態と限界を明らかにすることを目的として, 実運用ログを起点に複合指示ベンチマーク neoAI-InstructBench（図 1）を構築した. 本ベンチマークは, Closed-ended と Open-ended の双方を評価対象に含み, 5 つのカテゴリにまたがる制約を組み合わせた複合指示タスクとして構成した. また, 各指示がすべて異なる文面となるよう設計した. 主要なモデルを評価した結果, GPT-5.2 [5] のタスク完遂率は 67%であった. 加えて, 複合指示下における指示間干渉の傾向と, 表記・形式制約がボトルネックとなり得る点を分析した. さらに, 一部のモデルでは推論が停止せず最大トークン数に達することで無応答に至る失敗様式も確認した.

1) https://github.com/neoAI-inc/neoAI-InstructBench

表 1　5 つの指示カテゴリーと定義

カテゴリー名	定義	具体例
Format	構造, レイアウト, 記法	JSON 形式, 箇条書き, Markdown 禁止
Content	語彙, 事実情報	単語 A を N 回使用, 100 文字以内, A に関しては一切言及しない
Style	役割, トーン, 文体	情熱的な文体, 敬語, 「！」をつけて
Situation	状況理解, 推論, 判断	翻訳, A という前提を踏まえて, SWOT 分析を行う
Creativity	新規アイデア, 比喩	関係を日常生活に例えて説明, 造語生成

2　関連研究

LLM のアライメントにおいて, 指示追従能力の評価は核心的な課題であり, 多様な指示に対する追従性能をいかに定量化するかが議論されてきた. 多くの既存研究は Closed-ended 指示に基づく評価を採用し, 文字数制限や禁止ワード, フォーマット指定といった客観的なルールへの遵守率を指標としている. IfEval [1] は 25 種類, IfBench [3] は 58 種類の検証可能な指示から構成される. また, IFScale [4] は含めるべき単語数を 10 個から 500 個まで増やすことで指示の複雑性を制御しているが, これらはいずれも形式的な制御能力の測定に主眼を置いており, 正解が一意に定まらない自由記述や多様な回答を許容する指示の適切さを評価することが難しい.

一方で, 単一の指示だけでなく, 複数の指示を同時に満たす能力の評価も重要なトピックである. FollowBench [2] は, 同一カテゴリー内の指示を段階的に追加することで, 指示量が増加した際のモデルの挙動を検証している. また, 原田ら [6] は, 複数指示の成功率が個々の指示成功率の積で近似できることを示しており, 指示数の増加自体がタスク完遂の難易度を高める要因であることを示した.

また, 日本語特有の言語特性を考慮したベンチマークも提案されている. M-IfEval [7] は IfEval の多言語拡張であり, 日本語においても Closed-ended 指示の評価を可能にしている. また, ichikara-instruction2 に含まれるベンチマーク [8] は Open-ended 指示を含んでいるが, 定型化した指示表現が中心であり, 制約の組み合わせや表現の多様性は限定的である.

以上より, 既存研究は Closed-ended を中心とした指示追従の定量評価や, 複合指示に伴う難化傾向の理解を進めてきた一方で, 実運用で頻出する異種制約の同時要求を体系的に扱う評価枠組みは限定的である. そこで本研究では, 実運用ログに基づく複合指示ベンチマークを構築し, 実用環境における指示追従能力を体系的に評価する.

3　neoAI-InstructBench

3.1　データセット概要

本ベンチマークは, neoAI Chat [2] の社内利用ログからプライバシー情報を除去した発話をソースとしている. 実ログ由来の発話を起点とすることで, 実利用で生じるタスク目的と文脈の多様性を土台にしながら, 実運用から乖離しにくい複合指示を設計できる. 総数は 100 問, 計 326 指示であり, 1 タスクあたりの指示数は 2〜5 個とした（分布：2 個=34, 3 個=26, 4 個=20, 5 個=20）. また, 5 つのカテゴリー（詳細は後述）が, 指示数ベースで概ね均等となるように構成した.

3.2　構築プロセス

データの質と多様性を担保するため, LLM を補助的に活用した人手作成のアプローチを採用した. まず LLM を用いて指示案の候補を生成し, これを着想の起点として用いた. その後, 3 名の著者が実ログの文脈を反映させつつ, 候補を取捨選択しながら指示を複合化およびリライトし, 最終的な指示文を作成した. これにより, 定型的な生成では困難な, 文脈に依存した指示セットを構築した. また, 指示文の重複を排除し, 各指示がすべて異なる文面となるよう設計した. 最終的に, 作成者以外のメンバーによるクロスチェックを実施し, 指示の曖昧性を排除した.

3.3　指示カテゴリー

本ベンチマークでは, 指示を 5 つのカテゴリーに分類した（表 1）. なお, FollowBench で提案された 5 つのカテゴリーとの差分として, Example カテゴリーは, Few-shot 設定を前提としており自然なチャット文脈とは異なるため, 本研究では除外し, 実運用で求められることが多い Creativity を採用した.

2）https://neoai.jp/neoaichat

4 評価設定

4.1 評価指標

モデルの性能を多角的に測定するため, 2 つの指標を用いる. Prompt Acc は, 1 つのタスクに含まれるすべての指示を満たしたタスクの割合である. Inst. Acc は, 各指示の達成可否を判定し, その平均として算出する. いずれも成否判定に基づく.

4.2 評価手法

指示の性質に応じたハイブリッド評価を採用した. 文字数カウントや特定の単語の有無など, 指示カテゴリーに依らず客観的に検証可能な条件については, Python 関数によるルールベース評価（Closed-ended）を行った. 厳格な判定基準を採用し, 具体的には, モデル出力に含まれる装飾記号の除去やフォーマットの自動修正といった前処理を行わず, 生の出力文字列が条件を完全に満たしている場合のみを成功と判定した. 一方, ルールベースでの検証が困難な指示については, LLM-as-a-judge による評価（Open-ended）を行った. 評価モデルには判定根拠を出力させた上で, 遵守の可否のみを判定させた. 評価用プロンプトの詳細は Appendix C に示す.

4.3 実験環境

実験の再現性を担保するため, 以下の設定で推論を行った. 推論時の最大トークン数は 10,240 に設定し, Reasoning effort high 相当に設定した[3)]. また, 推論により全てトークンが消費され, 空のレスポンスが返された場合は最大 3 回リトライを行い, それでも空の場合は回答なしとして扱った. Throttling 等のサーバー側要因によるエラーは, このリトライの対象外とする.

5 実験結果

5.1 全体結果

主要な Closed model および Open weights model の評価結果を表 2 に示す. 全モデルの中で最も高い Prompt Acc を示したのは GPT-5.2 の 67%であり, neoAI-InstructBench の複合指示タスクの難易度の高さが示唆される. また, 指示正解率の平均 88%とタスク正解率の 67%の乖離は, 多くの指示を遵守していても, 特定の指示の欠落によりタスク全体が失敗となる場合があることを示す. さらに, Open weights モデルの Prompt Acc は最大でも 47%に留まり, 最大 67%の Closed model との間に差が見られた. 同系列の軽量モデルほど正解率が低下しており, 複合指示の同時達成には一定の推論・生成能力が必要である.

表 2 neoAI-InstructBench の評価結果

Model	Prompt Acc	Inst. Acc
GPT-5.2 [5]	**67.00**	**88.34**
GPT-5.1 [9]	66.00	87.12
GPT-5 mini [10]	61.00	75.15
GPT-5 nano [10]	35.00	58.59
Claude Opus 4.5 [11]	56.00	83.74
Claude Sonnet 4.5 [12]	51.00	81.60
Claude Haiku 4.5 [13]	39.00	74.23
Gemini 3 Pro [14]	51.00	81.60
Gemini 2.5 Flash [15]	45.00	77.30
Gemini 2.5 Flash Lite [15]	37.00	73.31
Kimi K2 Thinking [16]	47.00	79.75
Qwen3 235B A22B Instruct [17]	42.00	74.85
Qwen3 235B A22B thinking [17]	47.00	77.91
gpt-oss-120b [18]	45.00	79.75
gpt-oss-20b [18]	47.00	75.46
Gemma 3 27B Instruct [19]	20.00	62.27

6 分析と考察

6.1 指示数およびカテゴリーごとの傾向

指示数の増加に伴うタスク正解率（Prompt Acc）の推移を図 2 右に示す. 一般に, タスクに含まれる指示数が増えるほど全指示の同時充足は困難となり, Prompt Acc は低下する. 一方で, 一部のモデルでは指示数が増加する区間で一時的な上昇を含む非単調な変動も観測された. ただし, 指示数別に分割した場合のサンプル数が小さいため, 観測された変動には統計的な不確実性が含まれ得る.

次に, GPT-5.2 のカテゴリ別正解率（図 2 左）では, Situation（95.2%）および Creativity（96.9%）で高い正解率が得られた. 一方, Format（79.7%）や Content（83.6%）では相対的に失敗が多く, 複合指示下では形式的・語彙的な制約がボトルネックとなり得る. さらに, 文字種の制約や文末記号の統一といった表記指示でも失敗が観測され, 日本語固有の制約への適応には改善の余地がある. 例えば, 文末を全角の「？」または「！」で統一する Style 指示に対して半角の「!」を出力する, ひらがなと句読点のみを使用する指示に対して「同じれしぴでも」と漢字が

3) high が設定できない場合は 8,192 トークンを推論用に指定した.

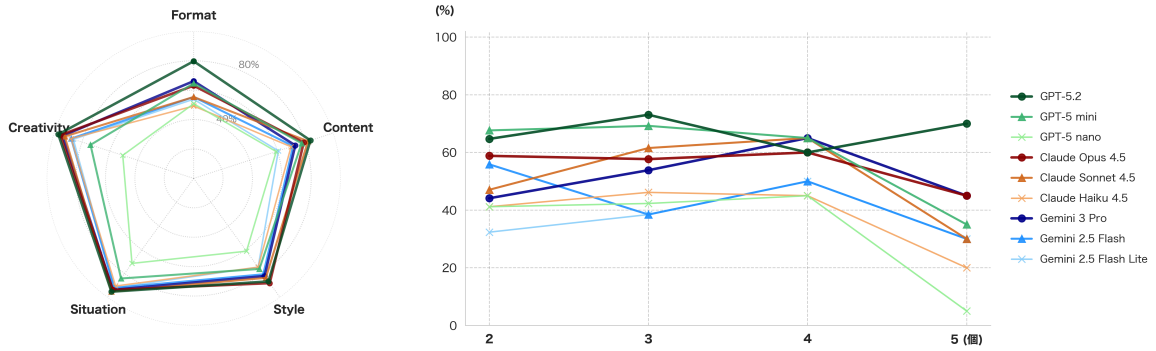

図 2　カテゴリー別正解率（左）と指示数ごとの正解率推移（右）

1 箇所だけ入り込む, といった例を確認した. 全モデルのカテゴリ別および指示数別の詳細な結果は Appendix A に示す.

6.2　指示間干渉と失敗様式

複合指示において, カテゴリー A の指示が含まれることがカテゴリー B の遵守に与える影響を評価するため, 干渉効果 $\Delta(A \to B)$ を定義する. 入力プロンプトを構成する指示の集合を $\mathscr{C}$ とし, 集合 $\mathscr{C}$ の条件下におけるカテゴリー B の正解率を $\mathrm{Acc}(B \mid \mathscr{C})$ とし, $A \in \mathscr{C}$ は $\mathscr{C}$ にカテゴリー A の指示が含まれることを表す. 干渉効果 $\Delta(A \to B)$ は, カテゴリー A の指示を含む場合と含まない場合のカテゴリー B の正解率差として次式で表される.

$$\Delta(A \to B) = \mathrm{Acc}(B \mid A \in \mathscr{C}) - \mathrm{Acc}(B \mid A \notin \mathscr{C}) \quad (1)$$

$\Delta(A \to B) < 0$ は, カテゴリー A の指示を含むことがカテゴリー B の性能を阻害していることを表す.

分析の結果, カテゴリ間で正負の干渉が観測された. 例えば, Content 指示が含まれる場合, Format 指示の正解率が約 10 ポイント低下する一方で, Style 指示の正解率は約 20 ポイント向上する傾向が確認された. 本分析は相関に基づくものであり, 厳密な因果関係の特定にはさらなる検証が必要である.

Overthinking による無応答　一部のモデルでは, 3 回のリトライを含む全ての生成において Reasoning のみで最大トークン数に達し, 回答が出力されない現象が確認された. 本実験では, GPT-5 nano で 26 件, GPT-5 mini で 11 件, GPT-5.1 で 2 件発生した. 複合指示が増えると回答探索に失敗し, 運用上致命的な無応答に至る可能性がある.

6.3　自動評価の課題

LLM を用いた自動評価の判定精度についても定性的に確認した. 評価基準は明文化しているものの,

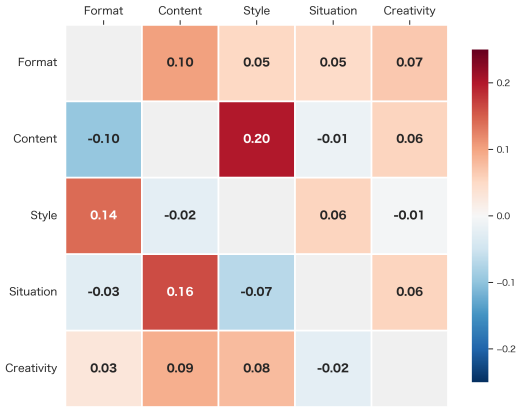

図 3　GPT-5.2 における指示カテゴリ間の干渉効果. 各セルは行の指示が存在することで列の指示の正解率がどの程度変化したかを示す. 値が負であれば干渉により精度が低下し, 正であれば精度が向上したことを意味する.

解釈が分かれるケースでは判定揺れが見られた. 例えば, 「Git 初心者にも心理的な不安を与えないように」という指示に対し, モデルが「『詰み』『禁じ手』『破壊光線』など, 取り返しのつかない失敗を連想させる語を用いた」として不合格と判定した事例があった. しかし, 当該指示は「不安を与えない」という要件自体に解釈の余地があり, 語彙選択が許容範囲かどうかの判断が一意に定まりにくい. 配慮や創造性の評価には難しさが残っており, 自動評価の限界として考慮する必要がある.

7　おわりに

本研究では, 実運用シナリオに基づく日本語複合指示ベンチマーク neoAI-InstructBench を構築し, 主要な大規模言語モデルを評価した. その結果, 最も高い Prompt Acc を示した GPT-5.2 においてもタスク正解率は 67%に留まり, 複合指示タスクの難易度の高さが示唆された. 今後は, 複合指示下で顕在化する干渉および表記・形式制約の破綻を抑制する手法の検討と, 評価の信頼性向上が課題である.

参考文献

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

[2] Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. FollowBench: A multi-level fine-grained constraints following benchmark for large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, **Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 4667–4688, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[3] Valentina Pyatkin, Saumya Malik, Victoria Graf, Hamish Ivison, Shengyi Huang, Pradeep Dasigi, Nathan Lambert, and Hannaneh Hajishirzi. Generalizing verifiable instruction following. **Advances in Neural Information Processing Systems**, Vol. 38, , 2025.

[4] Daniel Jaroslawicz, Brendan Whiting, Parth Shah, and Karime Maamari. How many instructions can llms follow at once? **arXiv preprint arXiv:2507.11538**, 2025.

[5] OpenAI. Introducing GPT-5.2: The most advanced frontier model for professional work and long-running agents. https://openai.com/index/introducing-gpt-5-2/, December 2025. Accessed: 2025-12-15.

[6] 原田憲旺, 山崎友大, 谷口仁慈, 小島武, 岩澤有祐, 松尾豊. 大規模言語モデルにおける複数の指示追従成功率を個々の指示追従成功率から推定する. 言語処理学会 第 31 回年次大会, 2025.

[7] Antoine Dussolle, A. Cardeña, Shota Sato, and Peter Devine. M-IFEval: Multilingual instruction-following evaluation. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, **Findings of the Association for Computational Linguistics: NAACL 2025**, pp. 6161–6176, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics.

[8] 堀尾海斗, 福田創, 小川隼斗, 鈴江万碧, 織田宥楽, 河原大輔. 日本語の包括的な指示追従性データセットの構築. 言語処理学会 第 31 回年次大会, 2025.

[9] OpenAI. GPT-5.1: A smarter, more conversational ChatGPT. https://openai.com/index/gpt-5-1/, November 2025. Accessed: 2025-12-07.

[10] OpenAI. GPT-5. https://openai.com/gpt-5/, 2025. Accessed: 2025-12-07.

[11] Anthropic. Claude Opus 4.5. https://www.anthropic.com/news/claude-opus-4-5, November 2025. Accessed: 2025-12-07.

[12] Anthropic. Claude Sonnet 4.5. Technical report, Anthropic, September 2025. Accessed: 2025-12-07.

[13] Anthropic. Claude Haiku 4.5. https://www.anthropic.com/news/claude-haiku-4-5, October 2025. Accessed: 2025-12-07.

[14] Google. A new era of intelligence with Gemini 3. https://blog.google/products/gemini/gemini-3/, November 2025. Accessed: 2025-12-07.

[15] Google DeepMind. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. **arXiv preprint arXiv:2507.06261**, 2025.

[16] Moonshot AI. Kimi K2 Thinking. https://moonshotai.github.io/Kimi-K2/thinking.html, November 2025. Accessed: 2025-12-07.

[17] Qwen Team. Qwen3 technical report. **arXiv preprint arXiv:2505.09388**, 2025.

[18] OpenAI. gpt-oss-120b and gpt-oss-20b model card. **arXiv preprint arXiv:2508.10925**, 2025.

[19] Gemma Team. Gemma 3. **Technical Report**, 2025.

表 3　全モデルにおけるカテゴリー別正解率および指示数ごとのタスク正解率 (%)

Model	Format	Content	Style	Situation	Creativity	2 Inst.	3 Inst.	4 Inst.	5 Inst.
GPT-5.2	79.7	83.6	86.6	95.2	96.9	64.7	73.1	60.0	70.0
GPT-5.1	80.0	81.8	85.1	95.2	93.8	64.7	69.2	70.0	60.0
GPT-5 mini	75.4	77.3	79.1	90.5	89.2	61.8	65.4	60.0	55.0
GPT-5 nano	58.5	60.6	62.7	76.2	73.8	38.2	38.5	30.0	30.0
Claude Opus 4.5	78.5	78.8	82.1	93.7	92.3	58.8	61.5	55.0	45.0
Claude Sonnet 4.5	76.9	75.8	79.1	90.5	89.2	55.9	57.7	50.0	40.0
Claude Haiku 4.5	69.2	71.2	73.1	85.7	83.1	44.1	42.3	35.0	30.0
Gemini 3 Pro	76.9	77.3	80.6	92.1	90.8	55.9	57.7	50.0	40.0
Gemini 2.5 Flash	72.3	74.2	76.1	87.3	86.2	50.0	50.0	45.0	35.0
Gemini 2.5 Flash Lite	67.7	69.7	71.6	82.5	80.0	41.2	42.3	35.0	30.0
Kimi K2 Thinking	73.8	75.8	77.6	88.9	87.7	52.9	53.8	45.0	35.0
Qwen3 Instruct	69.2	71.2	73.1	84.1	81.5	47.1	46.2	40.0	30.0
Qwen3 Thinking	72.3	74.2	76.1	87.3	84.6	52.9	53.8	45.0	35.0
GPT-OSS 120B	73.8	75.8	77.6	88.9	86.2	50.0	50.0	40.0	40.0
GPT-OSS 20B	69.2	71.2	73.1	84.1	81.5	52.9	50.0	45.0	40.0
Gemma 3 27B	55.4	57.6	59.7	73.0	69.2	23.5	23.1	15.0	15.0

A　全モデルの評価結果

表 3 にカテゴリー別および指示数ごとの正解率の詳細を示す.

B　データセット詳細例

以下に, 本ベンチマークに含まれるタスクの具体例を示す.

ID 1: PDF 変換ツール

ソース発話: "PDF を png に出来るだけ画質を落とさずに変換したい"
指示:

- **Situation**: Windows で行える無料のツールのみを使用してください。
- **Situation**: 対象の PDF は数百ページ以上あり、1 ページずつ手作業で変換するのは現実的ではありません。
- **Format**: 回答は番号付きリストで記述してください。
- **Content**: 画質に関する具体的な数値または設定名を少なくとも 1 つ含めて記述してください。

タスク: "PDF をできるだけ画質を落とさずに png に変換したいです。Windows で行える無料のツールのみを使用し、対象の PDF は数百ページ以上あり、1 ページずつ手作業で変換するのは現実的ではありません。回答は番号付きリストで記述し、画質に関する具体的な数値または設定名を少なくとも 1 つ含めて記述してください。"

ID 5: Python テストコード移行

ソース発話: "Python の pytest と factory_boy の関係について教えて。それぞれの範囲とか、同時に使えるのかとか"
指示:

- **Situation**: 既存の unittest ベースのテストコードが大量に存在するレガシープロジェクトでの段階的移行を前提として回答してください。
- **Format**: 回答は 3 行でお願いします。
- **Creativity**: 各段落の最初の文字を縦に読むと、『テスト』という言葉になるように文章を構成してください。
- **Style**: 敬語は一切使わないぶっきらぼうな口調で説明してください。

タスク: "Python の pytest と factory_boy の関係を unittest が大量に残るレガシープロジェクトでの段階的移行を前提に教えて。それぞれの範囲とか、同時に使えるのかとか。敬語なしのぶっきらぼうな口調を使い、回答は 3 行で、各行の頭文字を縦読みすると『テスト』になるように構成すること。"

C　Open-ended 評価プロンプト

あなたは厳格かつ一貫した採点者です。以下の情報に基づき、「評価対象の指示」に対して、「モデルの出力」が従っているかのみを評価してください。その他の指示は評価対象に含めません。

[元の依頼文]
{ソース発話}

[モデルに与えた最終プロンプト]
{タスク}

[評価対象の指示]
{評価対象の指示}

[モデルの出力]
{モデルの出力}

評価方針:
- 評価対象は上記の「評価対象の指示」に限定します。指示に直接関係しない観点は無視してください。
- 指示の意図を正しく汲み取り、「モデルの出力」がその意図に明確かつ十分に応えているかを判断してください。
- 部分的・曖昧・条件付きの遵守は 0 とし、明確に満たす場合のみ 1 とします。

出力要件:
1) 最初に「理由: 」で 1〜3 文の根拠を簡潔に述べる。
2) 最終行を必ず「点数: 1」(従っている) または「点数: 0」(従っていない) のみで記載する。
3) それ以外の出力や装飾は行わない。