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

原田 憲旺 [1]　山崎 友大 [2]　谷口 仁慈 [3]　小島 武 [1]　岩澤 有祐 [1]　松尾 豊 [1]
[1] 東京大学　[2] 京都大学　[3] 琉球大学
keno.harada@weblab.t.u-tokyo.ac.jp

概要

規定の文字数やフォーマットを守った文章生成や数千にも及ぶ条文からなる法律を遵守するなど，大規模言語モデルの更なる応用のため複数の指示追従性能は重要な側面である．複数の指示を同時に追従する性能の正確な推定ができると，未見の指示の組み合わせリスクのシミュレーションが可能となる．更に，その組み合わせの種類が膨大になるほどシミュレーションによるリスクの把握が重要性を増す．我々は複数の指示追従性能調査のためのベンチマーク ManyIFEval と StyleMBPP を作成し，同時に複数の指示追従する成功率は個々の指示の追従成功率の積で推定できるという経験則を得た．経験則により指示の未知の組み合わせに対して指示追従性能を推定できることを示した．また組み合わせる指示数が多くなればなるほど，同時に追従成功する可能性は劇的に低くなることを確認した．

1　はじめに

大規模言語モデルを活用することで様々な自然言語処理タスクを高性能に解かせることができる [1, 2]．タスクをモデルに解かせる際に，望む出力が得られるよう大規模言語モデルに指示出しを行うことはプロンプティングと呼ばれ，モデルが与えられた指示文に追従する性能 (instruction following)[3] は応用上重要である．

特に，複数の指示に全て追従する性能は大規模言語モデルの応用可能性を更に広げる上で重要であると考えられる．規定の文字数やフォーマットを守りつつ与えられたテーマについて文章を書く営みは，キャッチコピー生成や申請書作成など至るところで行われ，法律ドメインにおける応用では数千にも及ぶ条文からなる法律を遵守する必要がある．このように複数の指示・制約を全て守る行いは人間社会において求められるため，言語モデルの更なる応用のため複数の指示追従性能は重要な側面である．

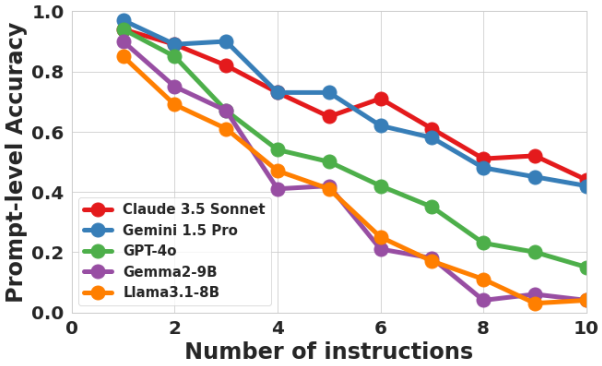

図 1　複数のモデルによる ManyIFEval の評価結果．指示数が多くなればなるほど，複数の指示を同時に追従成功するのは難しくなる．

しかしながら，大規模言語モデルの複数指示追従性能についての調査は十分に行われていない．既存の指示追従性能ベンチマークは言語モデルが様々な指示に対応できるか，指示の種類に対する性能の評価が主眼であった [4, 5]．複数指示を含むベンチマーク [6, 7] が近年提案されているものの，各指示数においてサンプルサイズが揃っておらず [6]，また指示追従の判定にモデルによる評価を用いる [7] ため信頼性に欠けるという課題がある．

我々は複数の指示追従性能調査のためのベンチマーク ManyIFEval と StyleMBPP を作成した．既存の指示追従性能を測るためのベンチマーク [7, 8, 4, 5] は同時に与えられる指示数は少数であるが，ManyIFEval は最大 10，StyleMBPP は最大 6 の指示数の追従性能が測定可能である．

作成したベンチマークを用いて複数のモデルを評価することにより，同時に複数の指示追従する成功率は個々の指示の追従成功率の積で推定できるという経験則を得た．得られた経験則によってある指示の組み合わせにおいて，同時に追従成功する見込みはどれくらいであるかを見積もることが可能であ

Are the weather conditions in the Arctic very cold most of the year?
Your response should follow the instructions below:
- The response must contain at least 3 placeholders represented by square brackets, such as [address].
- Do not include keywords ['no', 'yes'] in the response.

図 2　ManyIFEval のサンプル例.

You are an expert Python programmer, and here is your task: Write a python function to remove first and last occurrence of a given character from the string. Your code should pass these tests:
assert remove_Occ("hello","l") == "heo"
assert remove_Occ("abcda","a") == "bcd"
assert remove_Occ("PHP","P") == "H"
Your response should follow the instructions below:
- Ensure the file includes the MIT License notice
- Indent all code blocks using exactly two spaces; do not use tabs.

図 3　StyleMBPP のサンプル例.

る．また経験則は組み合わせる指示数が多くなればなるほど，それぞれの指示の追従が難しくなればなるほど同時に追従成功する可能性は劇的に低くなることを示唆しており，応用において重要な考慮事項である．

2　ベンチマーク

大規模言語モデルの複数の指示追従性能を計測するため，ManyIFEval と StyleMBPP を作成した．サンプル例は図 2，3 の通りである．評価の信頼性確保のため，プログラムによって客観的に検証が可能な指示で構成されている．ManyIFEval は IFEval[8] を，StyleMBPP は MBPP[9] を拡張して作成した．IFEval(Instruction-Following Eval) は*"Write a blog post about a trip to Japan."*といったような task prompt と，含めるべき・除外するべきキーワードの指定や，大文字・小文字の指定，文字数といった客観的に検証可能な指示群から構成されるベンチマークである．IFEval から 100 個の task prompt と 15 の指示を抽出し，1 つの task prompt あたり指示数 1 から 10 の指示からなるサンプルを作成し，合計 1000 サンプル作成した．

MBPP(Mostly Basic Programming Problems) は*"Write a Python function to sort the given arrayby using merge sort."*といった，python に関する基礎知識を問うようなコード生成ベンチマークである．MBPP は与えられたテストケースを正しくパスするかで評価がなされていたベンチマークであったが，MIT License 表記の追加，変数名の長さの制限や docstring の追加などの制約を加えたものが StyleMBPP である．MBPP のテストデータ 500 個それぞれに対して指示数 1 から 6 の指示からなるサンプルを作成し，合計 3000 サンプル作成した．

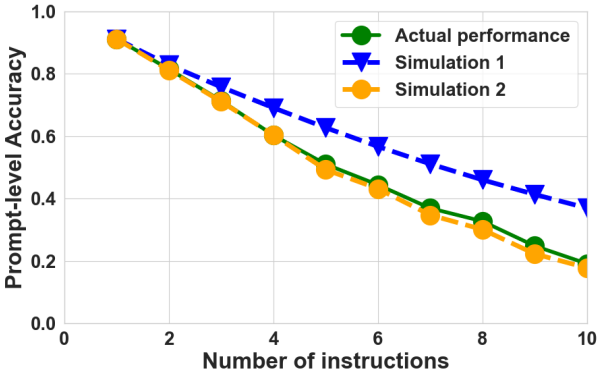

図 4　ManyIFEval における GPT-4o の実際の性能と経験則を用いた Simulation との比較．指示全てに追従する性能が，それぞれの指示の追従性能の積で見積もれることを確認.

指示の追従性能に関して，与えられた全ての指示に追従できたかどうかを以下の式 1 で定義される Prompt-level Accuracy という指標で表す．

$$\text{Prompt-level Accuracy (n)} = \frac{1}{m}\sum_{i=1}^{m}\prod_{j=1}^{n} s_i^j, \qquad (1)$$

ここで m はサンプルの数を，n は同時に与えられる指示数を表し，s_i^j はあるサンプル i の指示 j の追従が成功したかどうかをバイナリで表す．もしサンプル i の指示全ての指示に追従したら $\prod_{j=1}^{n} s_i^j = 1$ となり，それ以外の場合は 0 となる．

3　複数の指示追従成功率に関する経験則

3.1　実験結果から経験則を導出

ManyIFEval の評価結果は図 1 の通りであり，指示数が多くなればなるほど，複数の指示を同時に追従成功するのは難しくなることが観測された．評価に使用したモデルは GPT-4o (gpt-4o-2024-05-13) [1]，Claude 3.5 Sonnet(claude-3-5-sonnet-20240620)[10]，Gemini 1.5 Pro (gemini-1.5-pro-002) [2]，Gemma2 (gemma-2-9b-it) [11]，Llama 3.1 (Meta-Llama-3.1-8B-Instruct) [12] である．言語モデルの推論の際にはゼロショットプロンプティングの設定で，

GPT 以外のモデルは greedy decoding で行い，GPT は nucleus sampling のパラメータである top-p を 1e-10 として行った．

指示の組み合わせと，複数指示の同時追従成功率との関係性について考察するために，個々の指示の追従成功率から複数指示の同時追従成功率を推定を試みた．各指示を x_i とし，n 個の命令が同時に与えられたときに命令 x_i に従うことができる確率を $\mathrm{success}(x_i, n)$ とする．また，$X(x_1, x_2, \ldots, x_{n-1}, x_n)$ をプロンプトに含まれる指示の集合とし，$P(X)$ を与えられた全ての指示に追従成功する確率とする．図 4 で示すように，実験によって経験的に，n が増加すると $\mathrm{success}(x_i, n)$ は $\mathrm{success}(x_i, 1)$ より低くなること，そして $P(X)$ は以下のように推定できることを確認した．

Simulation1:

$$P(X) = \mathrm{success}(x_1, 1) \times \mathrm{success}(x_2, 1) \times \cdots \times \mathrm{success}(x_n, 1)$$

Simulation2:

$$P(X) = \mathrm{success}(x_1, n) \times \mathrm{success}(x_2, n) \times \cdots \times \mathrm{success}(x_n, n)$$

この経験則が示唆するところは，指示全てに追従する性能が，それぞれの指示の追従性能の積で見積もれることである．指示の追従成功率が 0.5 である指示が 2 つ，追従成功率が 0.9 である指示が 2 つあるとする．4 つ全ての指示に追従する性能は，指示の中で一番低い成功率の 0.5 になるのではなく，Simulation 1 で確認したように $0.5 \times 0.5 \times 0.9 \times 0.9 = 0.2025$ 程度になるだろうと見積もれる．またより高い精度で推定を行える Simulation 2 で確認したように，それぞれの指示の追従成功率は同時に与えられる指示によって影響を受けることを考慮すると，Simulation 1 の結果より悪くなると期待される．

3.2 経験則の汎用性確認 (ManyIFEval 内の指示の未知の組み合わせ)

経験則の確認として，指示の未知の組み合わせに対して，経験則を活用することで同時に追従成功する確率の推定を試みる．ManyIFEval の指示から 10 個取り出し，10 個の中で 3 つの指示の組み合わせ合計 120 通り作成した．120 通りから訓練用の組み合わせを 60 通り，テスト用の組み合わせに分割した．task prompt は訓練・テスト共通とする．

それぞれの指示が 1 つだけ与えられた時の追従成功率である $\mathrm{success}(x_i, 1)$ は 10 個の指示全てに関して実験を行い算出する．GPT-4o (gpt-4o-2024-05-13) による算出結果は表 1 の通りである．訓練用の組み合わせのみを用いて，3 個の命令が同時に与えられたときに命令 x_i に従うことができる確率である $\mathrm{success}(x_i, 3)$ の算出を行い，テスト用の指示の未知の組み合わせの同時追従成功確率を推定する．テスト用の組み合わせの実際の Prompt-level Accuracy は 0.619，Simulation 1 による推定結果は 0.652，訓練用の組み合わせを用いた Simulation 2 の推定結果は 0.609 となり，指示の未知の組み合わせに対して，経験則を活用することで同時に追従成功する確率の推定できることを確認した (表 2)．

表 1 ManyIFEval における単一指示の追従成功率

指示	追従成功率
含めるべきキーワード	0.98
キーワードの出現回数	0.96
禁止キーワード	0.95
特定の文字の出現回数	0.62
文字数	0.81
文章数	0.73
パラグラフ数	0.87
placeholder	0.95
箇条書き	0.88
タイトル	1.0
大文字指定	0.97
小文字指定	0.97
大文字単語の数	0.87
引用符	1.0
コンマ禁止	1.0

表 2 ManyIFEval における指示の未知の組み合わせに対して，実際の結果と経験則による推定値の比較．

項目	実測値	推定値
Prompt-level Acc	0.619	-
Simulation 1	-	0.652
Simulation 2	-	0.609

3.3 経験則の汎用性確認 (StyleMBPP での確認)

経験則がドメイン外のベンチマークにおいても同様に観測されるか確認するために，StyleMBPP で実験を行った．StyleMBPP は指示が 6 つ用意されており，それぞれの指示の追従成功率を求めたのちに，6 つの指示全てに追従する性能を，個々の指示の追従性能の積で推定する．モデルは GPT-4o (gpt-4o-mini-2024-07-18) を使用し実験を行なった．

StyleMBPP で最も追従成功率が高い指示は「MIT

表 3　StyleMBPP における単一指示の追従成功率

指示	追従成功率
MIT License 表記	1.000
インデント数	1.000
関数の docstring	1.000
条件比較	0.992
一行当たりの文字数	0.868
変数名の長さ	0.794

表 4　StyleMBPP における複数指示の実測値と推定結果

項目	実測値	推定値
Prompt-level Acc	0.660	-
Simulation 1	-	0.684
Simulation 2	-	0.679

License 表記」,「インデント数」,「関数の docstring」であり，指示が 1 つだけ与えられた際の追従成功率は 500 件中 500 件成功の 1.0 である．最も追従成功率が低い指示は「変数名の長さ」であり指示が 1 つだけ与えられた際の追従成功率は 500 件中 500 件成功の 0.794 である．算出結果は表 3 の通りである．

Prompt-level Accuracy は 0.660，それぞれの指示が 1 つだけ与えられた際の追従成功率のみを用いた Simulation 1 による推定結果は 0.684 となった．6 つの指示全てに追従する性能が個々の指示の追従性能の積でおおよそ推定できたことを確認した (表 4)．

StyleMBPP は追加した指示の追従成功に関して客観的に検証可能なだけでなく，与えられた task prompt の要求を満たせたかの検証がテストケースの確認によって行える．何も追加の指示が与えられていない task prompt のみの task prompt 要求成功率は 0.76 であり，指示が 1 の場合は 0.75，指示が 6 の場合は 0.72 であった．Simulation 2 の算出で考慮に入れた，指示数が増えると各指示の追従成功率が下がるという現象が task prompt レベルでも確認された．

3.4　経験則の考察

同時に複数の指示追従する成功率は個々の指示の追従成功率の積で推定できるという経験則は，指示の追従成功に関しての独立性を示唆する．今回 ManyIFEval や StyleMBPP で採用した指示群は，ある指示を追従した場合には他の指示の追従が不可能となるような，互いに矛盾するような指示を取り除いたことが独立性示唆の一因と考えられる．

また今回対象としたベンチマークで扱う指示群では，ある時点のトークン予測において，考慮すべき指示は限られるという特性も独立性示唆の一因と考えられる．例えばコード生成において「MIT License 表記」は全トークン系列の最初の部分で達成が可能であり，変数の定義を行う際に「変数名の長さ」の指示を考慮したトークン予測を行えば良い．プロンプトレベルでは同時に課されている指示であっても，ある時点のトークン予測レベルで見ると考慮すべき指示は限られるため，独立性が示唆されるというわけである．

大規模言語モデルのモデル構造であるトランスフォーマー [13] の特徴から今回の経験則への考察も今後の課題である．トランスフォーマーで採用されている注意機構はトークン系列全体に合計 1 となるように重みづけを行って注目部分を取捨選択しているが，Simulation 2 の算出で考慮に入れた，指示数が増えると各指示の追従成功率が下がるという現象は注意機構との関係性で今後分析予定である．またある時点のトークン予測において，考慮すべき指示は限られるという特性も注意スコアをもとに裏付けが行える可能性があり今後の研究課題である．

4　おわりに

本論文では大規模言語モデルの更なる応用のために重要である複数の指示追従性能に着目し，同時に複数の指示追従する成功率は個々の指示の追従成功率の積で推定できるという経験則を得た．我々が作成したベンチマークを活用することによって経験則の汎用性に関して ManyIFEval 内の指示の未知の組み合わせにおける検証，ドメイン外である StyleMBPP における検証によって確認した．同時に複数の指示追従する成功率は個々の指示の追従成功率の積で推定できるという経験則に関して，扱う指示の特徴・モデル構造やトークン予測の観点から考察を行い今後の研究の方向性を示した．

得られた経験則を活用することで指示の未知の組み合わせに対して指示追従性能を推定できることは応用上有用であり，また組み合わせる指示数が多くなればなるほど，同時に追従成功する可能性は劇的に低くなることを示唆しており，応用において重要な考慮事項である．

参考文献

[1] OpenAI. Gpt-4o system card, 2024.

[2] Gemini Team. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context, 2024.

[3] Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In **International Conference on Learning Representations**, 2022.

[4] Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, **Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing**, pp. 5085–5109, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[5] Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Alpacaeval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval, 5 2023.

[6] Bosi Wen, Pei Ke, Xiaotao Gu, Lindong Wu, Hao Huang, Jinfeng Zhou, Wenchuang Li, Binxin Hu, Wendy Gao, Jiaxin Xu, et al. Benchmarking complex instruction-following with multiple constraints composition. **arXiv preprint arXiv:2407.03978**, 2024.

[7] 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.

[8] 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.

[9] Augustus Odena, Charles Sutton, David Martin Dohan, Ellen Jiang, Henryk Michalewski, Jacob Austin, Maarten Paul Bosma, Maxwell Nye, Michael Terry, and Quoc V. Le. Program synthesis with large language models. In **n/a**, p. n/a, n/a, 2021. n/a.

[10] Anthropic. Claude 3.5 sonnet, 2024.

[11] Gemma Team. Gemma 2: Improving open language models at a practical size, 2024.

[12] Llama Team. The llama 3 herd of models, 2024.

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, **Advances in Neural Information Processing Systems**, Vol. 30. Curran Associates, Inc., 2017.