Web エージェントにおけるヒントの抽象度とタスク関連度が性能に与える影響の分析

小原涼馬 [1]　秋元康佑 [1]　榎本昌文 [1]　粟井響生 [2*]
竹岡邦紘 [1]　原口大地 [1]　田村拓也 [1]　小山田昌史 [1]
[1]NEC データサイエンスラボラトリー
[2] 大阪大学 理学研究科
{ryoma-obara,kosuke_a,masafumi-enomoto}@nec.com
u166277g@ecs.osaka-u.ac.jp
{k_takeoka,daichi-haraguchi,tamura-takuya}@nec.com
oyamada@nec.com

概要

LLM を用いて Web タスクを遂行するエージェントにおいて，過去に実行したタスクから有用な知見（ヒント）を抽出し，タスクの実行に活用するアプローチが注目されている．既存研究では，実行するタスクとヒントとして利用したい過去のタスクの関連度に関わらずヒントの抽象度が固定的である．本研究ではヒントの抽象度とタスク間の関連度の組み合わせがエージェント性能にどのように影響するか体系的に分析する．WebArena Mod ベンチマークを用いて評価を行った結果，関連度の高いタスク間では具体的な実行情報を含むヒントが最も高い性能向上を示す一方，関連度が低い場合は，抽象化されたヒントが高い性能を示すことが確認された．

1　はじめに

大規模言語モデル（LLM）の発展により，Web や GUI 環境で複雑なタスクを遂行するエージェントの研究が急速に進展している [1]．これらのタスクは，単一の行動ではなく，複数ステップにわたり適切に行動をする必要がある．一般的な設定では，エージェントは実行中のタスク（以下，ターゲットタスク）において環境とインタラクションを行いながら，試行錯誤を通じてタスク遂行に必要な知識を動的に獲得する．対象となる Web サイトや UI は多様であり，個々のタスク遂行に必要な知識を LLM が事前学習によって必ずしも獲得しているとは限らない．そのため，エージェントは実行時のインタラクションに依存した試行錯誤によってタスクを遂行せざるを得ない．しかし，このような試行錯誤は非効率であり，結果として失敗率も依然として高いという課題がある [2]．

効率および性能改善のため，過去にエージェントや人間が実行したタスク（以下，ソースタスク）の軌跡から，有用な知識をあらかじめ抽出・保存し，これをターゲットタスクの実行時にプロンプトに含めて活用するアプローチが近年注目されている [3]．これにより，過去のタスク実行から得られた経験知識を活用できるとともに，LLM が事前学習では十分に獲得できない Web 固有の操作知識や戦略を実行時に補完できるという利点がある．本研究では，このようにタスク遂行に有用な知識として保存・提示される情報を「ヒント」と呼ぶ．

既存研究ではヒントを抽出する際，タスク固有のエンティティや入力値への依存を減らすために情報を抽象化することで，タスク間での汎化性能が向上し，結果としてエージェント性能が改善されることが示されている [4, 5, 6, 7]．例えば，具体的なエンティティを除去して手順情報のみを抽出するものから，手順情報自体を含めず，より汎用性の高い知見を抽出するといった抽象化が行われている．

この時，適切なヒントの抽象度は、ソースタスクとターゲットタスクの関連度に依存すると考えられる。例えば，タスク間の関連度が高い場合には具体的な実行情報を含むヒントが有効である一方，関連度が低い場合には，より抽象化されたヒントが有効になる可能性がある．しかし，既存研究ではヒントの抽象度を固定した上で評価を行うことが多く，

* 本論文への貢献は NEC へのインターンシップ活動中に行われたものである．

ソースタスクとターゲットタスクの関連度に応じて，どのようなヒント抽象度が有効となるかについては明示的に分析されていない．

本研究では，「ソースタスクとターゲットタスクの関連度に応じて，ヒントの抽象度の効用はどのように変化するのか」を検証する．具体的には，3 種類の抽象度のヒントと 4 段階の関連度の組み合わせがエージェント性能に与える影響を実験的に分析する。WebArena Mod ベンチマークを用いた実験の結果，タスク間で手順やエンティティなどの共通する情報を持つ場合は，対応する手順およびエンティティに関する情報を保持することが有効な一方，タスク間の関連度が極めて低い場合は汎用的な知見ヒントが比較的高い性能であることが確認された．さらに，タスク毎の成功数変化を分析したところ，ヒントの条件によって正の影響と負の影響の傾向が異なることが確認された．

2 関連研究

過去のタスク実行から得られた知識を再利用し，LLM エージェントの性能向上を図る研究が近年活発に行われている．Learn-by-Interact[8] は，過去のデモンストレーションや軌跡をそのまま検索・再利用する枠組みを提案したが，生のトラジェクトリは長くノイズを含むため，タスク間での転移性が低い．この課題に対し，AutoGuide[4] は対照的な軌跡ペアからガイドラインを抽出することで，より抽象化された再利用可能知識を得られることを示した．Agent Workflow Memory[5] や Memp[6] では，軌跡をそのまま保持する記憶と，高レベルな手続きとして要約した記憶を区別して扱うことで，タスク横断的な知識の再利用を可能にしている．さらに，ExpeL[6] や JEF HINTER[7] は，成功・失敗軌跡を対比・集約することで，特定の手順に依存しない抽象的な知見を自然言語ヒントとして抽出し，複数の Web ベンチマークにおいて一貫した性能向上を報告している．

しかし，これらの研究はヒントの抽象度を固定した上で評価が行われており，ソースタスクとターゲットタスクの関連度とヒントの抽象度の組み合わせについての分析は十分にされていない．

3 実験設定

本研究では，WebArena[2] を用いて実験を行う．WebArena は，実在の Web サイトを模した複数の環境上で，検索，クリック，フォーム入力などの逐次的な Web 操作を行うエージェントの性能を評価するためのベンチマークである．WebArena では複数のサイトドメインが用意されており，各タスク毎に使用するサイトドメインが決められている．

3.1 ヒント設計

本研究では，(1) ターゲットタスクに対してどのような関連度のソースタスクを選択するか (2) 選択されたソースタスクからどのような抽象度のヒントを生成するかの二軸に着目し，その相互作用を分析する．

タスク間関連度の定義とソースタスク選択方法 本研究では，ソースタスクとターゲットタスクの関連度を，タスク記述文の類似度と，WebArena のサイトドメイン情報を用いて定義する．具体的には，以下の 4 条件を設定し，各条件ごとにヒント生成に用いるソースタスクを一意に選択する．

- **同一タスク**：ターゲットタスクと完全に同一のタスクをソースタスクとして用いる．過去に実行したタスクの知識をそのまま再利用できる理想的な条件に相当する．
- **高関連タスク**：ターゲットタスクと同一のサイトドメインのタスクの中から，タスク記述文の BM25 スコア[1)]が最大となるタスクを選択する．WebArena には同一の操作手順を要するが異なるエンティティを対象としたタスクが複数存在するため，この条件では手順レベルの知識の転移が期待される．
- **低関連タスク（同一ドメイン）**：ターゲットタスクと同一のサイトドメインに属するタスクの中から，BM25 スコアが最小となるタスクを選択する．実行手順の共通性は低い一方で，同一ドメインに由来する Web サイト特有の操作知識や UI 構造に関する知見の転移が期待される．
- **極低関連タスク（異ドメイン）**：ターゲットタスクとは異なるサイトドメインに属するタスクの中から，BM25 スコアが最小となるタスクを選択する．タスク手順やサイト固有の操作知識の転移はほとんど期待できないが，Web タスク一般に共通する汎用的な知見の再利用が期待される．

ヒントの抽象度 本研究では，以下の 3 段階の抽

1) https://pypi.org/project/rank-bm25/

象度を定義する．

- **具体手順ヒント**：タスク内で登場する具体的なエンティティを明示し，実行手順を記述する．
- **抽象手順ヒント**：具体的なエンティティを抽象化し，操作の流れや手順構造のみを記述する．
- **汎用知見ヒント**：特定タスクに依存せず，複数タスクに共通する注意点や戦略を記述する．

ヒント生成に用いる実行軌跡には，WebArena のタスクに対して人間による成功軌跡が付与された WonderBread[9] のデータを用いる．これらの軌跡を入力として，大規模言語モデルによりヒントを生成する．ヒント生成には `gpt-oss-120b`[2] を用い，ヒントの抽象度を制御するため，種類ごとに異なるプロンプトを用いる．プロンプトの詳細は付録 A に示す．

3.2 評価設定

エージェントの性能評価には WebArena Mod[10] を用いる．WebArena Mod は既存の WebArena を基に，一部の評価設定および環境仕様を修正したものであり，評価の妥当性向上を目的として設計されている．評価対象タスクとして WebArena Mod のうち，WonderBread において人手によりタスクの正確性が確認された 164 タスクを用いる．エージェントは最大 30 ステップ以内にタスク成功条件を満たした場合に成功と判定される．各ヒント条件で 3 回試行し，平均スコアを算出する．

エージェント設定　Web エージェントの評価・実装を行うためのフレームワークである AgentLab[11] 上で提供されている GenericAgent を使用する．GenericAgent は ReAct[12] 型の逐次意思決定エージェントであり，各ステップ t において，タスク指示，これまでの行動履歴，および現在の観測 o_t を含むプロンプトを入力として，次の行動 a_t を生成する．エージェントの基盤モデルとして `gpt-oss-120b` を用い，ハイパーパラメータは，すべての条件で共通に固定する．

ヒントの組み込み方法　ヒントは生成されたヒント文字列（{hint_str}）を以下に示す形式でエージェントのプロンプトの先頭に挿入することによってエージェントに組み込む．ヒント文字列はタスク実行中の全ステップで同じである．ベースラインとしてヒントなし条件を設定し，「`no hint provided`」というヒント文字列を利用する．

2） https://huggingface.co/openai/gpt-oss-120b

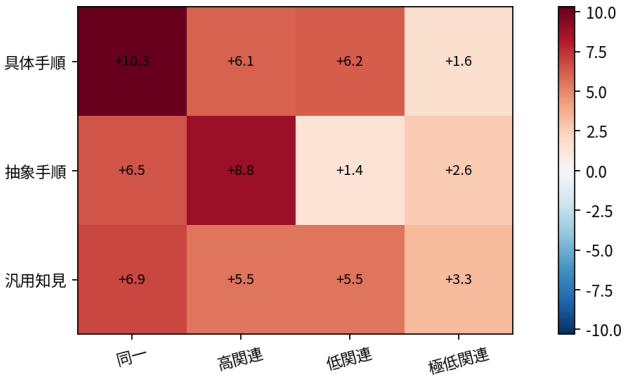

図 1: ヒントの抽象度（具体手順 / 抽象手順 / 汎用知見）とヒントソースタスクとの関連度ごとの性能を示すヒートマップ．各セルはヒントを用いない場合に対する平均成功率の差分を表し，正の値ほど性能向上が大きいことを示す．

```
<tips>
The following information contains tips from human
    experts that may help you solve this task.
{hint_str}
</tips>
```

4 実験結果

図 1 に，ヒント抽象度およびタスク関連度ごとのヒントなし条件に対する平均成功率改善幅を示す．実験結果について，ヒント抽象度を固定してタスク間関連度方向に見る視点と，タスク間関連度を固定してヒント抽象度方向に見る視点の両方から論じる．

ヒント抽象度 (縦軸) を固定した場合　具体手順ヒントでは，同一タスク条件において+10.3 ポイントの成功率向上が確認された．一方で，タスク関連度が低下するにつれて向上幅は小さくなり，異なるサイトドメインに属する極低関連条件では +1.6 ポイントにとどまった．

抽象手順ヒントでは，同一タスクおよび高関連タスクにおいて成功率の向上が確認されたが，低関連条件では性能向上は限定的であった．

汎用知見ヒントは，全ての関連度条件において小幅ながら正の性能差分を示した．

タスク関連度 (横軸) を固定した場合　同一タスク条件では，具体手順ヒントが最も高い成功率を示した．一方，タスク関連度が低下するにつれて，具体手順ヒントの成功率は低下した．

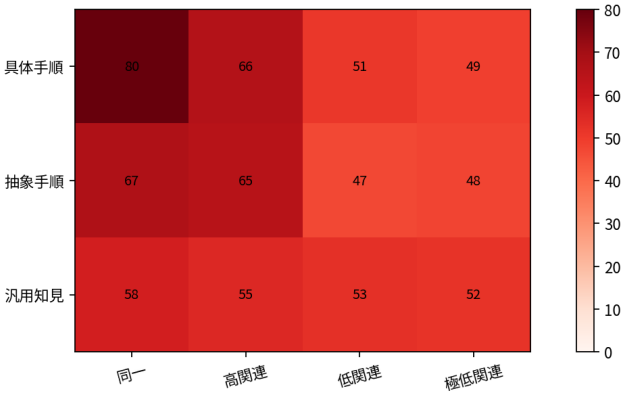

図 2: ヒント抽象度およびタスク関連度ごとの改善数の増加（ヒントなし条件との差分）．各タスクについて 3 回試行した際の成功回数の差分が正となった場合に，それらを合算した値を示す．

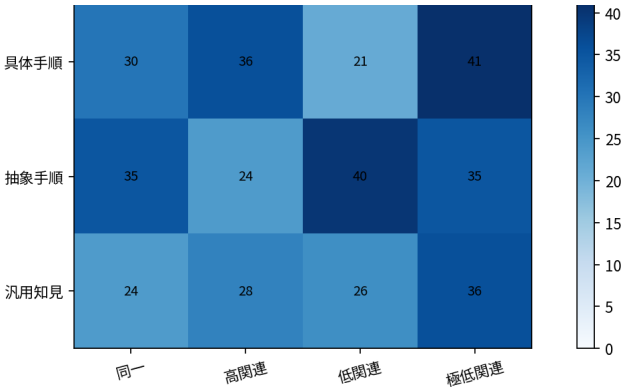

図 3: ヒント抽象度およびタスク関連度ごとの悪化数の増加（ヒントなし条件との差分）．各タスクについて 3 回試行した際の成功回数の差分が負となった場合に，その絶対値を合算した値を示す．

低関連条件および極低関連条件では，汎用知見ヒントが他のヒント抽象度と比較して相対的に高い成功率を示した．

5 分析

本節では，成功率変化の要因を詳細に分析するため，各タスクごとのベースラインからの成功回数の増加分（改善数）と減少分（悪化数）それぞれの合計に着目して考察を行う．まず，図 2 に示す改善数に着目すると，同一タスクおよび高関連タスクにおいて，具体手順ヒントおよび抽象手順ヒントでは改善数が 65〜80 件と，他の条件と比べて多くの改善が確認された．特に同一タスク条件下で具体手順ヒントを与えた場合は改善が著しかった．この結果は，タスク間の関連度が手順さらにはエンティティを共有するほど高い場合には，対応する手順およびエンティティに関する情報をヒントに保持することが有効であることを示唆している．

一方で，低関連および極低関連条件の間の差に注目すると，改善数はどの抽象度でも大きな差が見られなかったが、図 3 に示す悪化数は具体手順と汎用知見について極低関連条件で相対的に顕著な悪化が確認された．この結果は，サイトドメインが大きく異なるほどタスク関連度が低い状況では，無関係な情報がヒントとして与えられることでエージェントの行動に悪影響を及ぼし，ヒントの抽象度に関わらず性能低下につながった可能性を示唆している．

興味深い点として，以上の分析結果をまとめると，ヒントを与えることによる成功率への正の効果はヒントの抽象度とタスク共通要素の一致関係に強く影響を受ける一方で，負の効果はタスクドメインの差に特に影響を受けることを示唆している．このことから，今後ヒントの効果を左右する要因を分析する際には，ヒントの正の影響と負の影響について分けて分析する必要があると考えられる．

しかし，抽象手法・低関連度条件の組み合わせについては特に悪化数の観点で上記の傾向に合致していないため，今後さらなる実験・分析が求められる．

6 まとめ

本研究では，LLM を用いて Web タスクを遂行するエージェントにおいて，ヒントの抽象度とヒント生成に用いるタスクと実行するタスクの関連度の組み合わせが，エージェント性能に与える影響を定量的に分析した．実験の結果，タスク間の関連度が手順さらにはエンティティを共有するほど高い場合には，対応する手順およびエンティティに関する情報を保持することが有効な一方，タスク間の関連度が極めて低い場合は汎用的な知見ヒントが比較的高い性能であることが確認された．また，タスク毎の成功数変化の分析により，ヒントの条件によって正の影響と負の影響の傾向が異なることが確認され，ヒントの効果にはより詳細な分析が必要なことが示唆された．

参考文献

[1] Liangbo Ning, Ziran Liang, Zhuohang Jiang, Haohao Qu, Yujuan Ding, Wenqi Fan, Xiao-yong Wei, Shanru Lin, Hui Liu, Philip S. Yu, and Qing Li. A survey of webagents: Towards next-generation ai agents for web automation with large foundation models. In **Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2**, KDD '25, p. 6140–6150, New York, NY, USA, 2025. Association for Computing Machinery.

[2] Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. Webarena: A realistic web environment for building autonomous agents. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun, editors, **International Conference on Representation Learning**, Vol. 2024, pp. 15585–15606, 2024.

[3] Zeyu Zhang, Quanyu Dai, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. A survey on the memory mechanism of large language model-based agents. **ACM Trans. Inf. Syst.**, Vol. 43, No. 6, September 2025.

[4] Yao Fu, Dong-Ki Kim, Jaekyeom Kim, Sungryull Sohn, Lajanugen Logeswaran, Kyunghoon Bae, and Honglak Lee. Autoguide: automated generation and selection of context-aware guidelines for large language model agents. In **Proceedings of the 38th International Conference on Neural Information Processing Systems**, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc.

[5] Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory. In **Forty-second International Conference on Machine Learning**, 2025.

[6] Runnan Fang, Yuan Liang, Xiaobin Wang, Jialong Wu, Shuofei Qiao, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. Memp: Exploring agent procedural memory, 2025.

[7] Hadi Nekoei, Aman Jaiswal, Patrice Bechard, Oleh Shliazhko, Orlando Marquez Ayala, Mathieu Reymond, Massimo Caccia, Alexandre Drouin, Sarath Chandar, and Alexandre Lacoste. Just-in-time episodic feedback hinter: Leveraging offline knowledge to improve llm agents adaptation, 2025.

[8] Hongjin SU, Ruoxi Sun, Jinsung Yoon, Pengcheng Yin, Tao Yu, and Sercan O Arik. Learn-by-interact: A data-centric framework for self-adaptive agents in realistic environments. In **The Thirteenth International Conference on Learning Representations**, 2025.

[9] Michael Wornow, Avanika Narayan, Ben Viggiano, Ishan S. Khare, Tathagat Verma, Tibor Thompson, Miguel Angel Fuentes Hernandez, Sudharsan Sundar, Chloe Trujillo, Krrish Chawla, Rongfei Lu, Justin Shen, Divya Nagaraj, Joshua Martinez, Vardhan Agrawal, Althea Hudson, Nigam H. Shah, and Christopher Ré. Wonderbread: a benchmark for evaluating multimodal foundation models on business process management tasks. In **Proceedings of the 38th International Conference on Neural Information Processing Systems**, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc.

[10] Haochen Zhang, Masafumi Enomoto, Ryoma Obara, Kunihiro Takeoka, and Masafumi Oyamada. Webarena mod, 2025.

[11] Thibault Le Sellier de Chezelles, Maxime Gasse, Alexandre Lacoste, Massimo Caccia, Alexandre Drouin, Léo Boisvert, Megh Thakkar, Tom Marty, Rim Assouel, Sahar Omidi Shayegan, Lawrence Keunho Jang, Xing Han Lù, Ori Yoran, Dehan Kong, Frank F. Xu, Siva Reddy, Graham Neubig, Quentin Cappart, Russ Salakhutdinov, and Nicolas Chapados. The browsergym ecosystem for web agent research. **Transactions on Machine Learning Research**, 2025. Expert Certification.

[12] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. 2023. Publisher Copyright: © 2023 11th International Conference on Learning Representations, ICLR 2023. All rights reserved.; 11th International Conference on Learning Representations, ICLR 2023 ; Conference date: 01-05-2023 Through 05-05-2023.

A　ヒント生成に用いたプロンプト

共通プロンプト

```
You are an expert web task agent coach.
Below are ONLY successful (positive) step-by-step action
     logs for solving the task.
{log_str}
```

{log_str}は文字列化されたアクション軌跡である．

具体手順ヒント生成プロンプト

```
From ONLY these successful runs, build a **concrete,
     chronological workflow** that another agent can
     replay.
The workflow should retain the specific entities and
     wording seen in the logs (element IDs/names, button
     labels, field names, table entries, URLs, search
     terms, numbers) whenever they appear. Do not invent
     new names.
Requirements:
- Keep the exact sequence in executable steps with sub-
     points.
- Use explicit browser actions (click, type, select,
     scroll, wait, verify, read).
- Include wait/verification cues when pages change or
     results update.
- If the logs show branching/validation, keep the
     condition and the chosen branch.
- If a needed label/value is not present in the logs,
     state the action generically without fabricating
     content.
Output format:
- Numbered steps (1., 2., 3., ...), each with short sub-
     bullets.
- No intro or —conclusiononly the workflow.
```

抽象手順ヒント生成プロンプト

```
From ONLY these successful runs, build an **abstract,
     reusable workflow** another agent can follow for
     similar tasks.
Abstraction Policy (VERY IMPORTANT):
- DO abstract task-dependent values into natural language
      descriptions.
- DO NOT abstract UI labels that remain stable across
     tasks: menu names, section names, button labels,
     table headers, etc.
- DO NOT provide concrete examples of any values. This
     includes:
    - NO sample dates.
    - NO sample numbers.
    - NO example names.
    - NO sample search keywords.
    - NO 'e.g.' or 'for example' clauses.
    - NO invented placeholder values.
- Abstract values MUST be expressed ONLY as conceptual
     descriptions such as
    - enter the required date
    - type the intended search keyword
    - select the relevant option
WITHOUT giving explicit examples or sample values.
Requirements:
    - Preserve the working order and decision-making
        structure from the successful runs.
    - Use precise action verbs (click, type, select,
        scroll, wait, confirm, read, extract).
    - Explain where to locate elements (navigation bar,
        submenu, table rows, filter inputs, buttons) and
        how to confirm state changes.
    - Keep steps concise and executable.
    - When referring to abstract values, describe them
        naturally and conceptually.
Output format:
    - Numbered steps (1., 2., 3., ...), each with short
        sub-bullets.
    - No introductory or closing text only the workflow.
```

汎用知見ヒント生成プロンプト

```
From ONLY these successful runs, distill **actionable
     insights and cautions** that help an agent solve
     similar tasks.
    These must be standalone tips, not a workflow.
    Requirements:
    - Provide situation-based Do/Be careful insights (
        element finding, timing, verification, navigation
        , data reading).
    - Avoid sequential language (first, next, then) and
        avoid multi-step chains.
    - Keep references abstract focus on behaviors,
        triggers, and checks rather than specific labels.
    - Each insight must be locally applicable and
        independent.
    Output format:
    - Numbered list of 8-15 insights.
    - No intro or —conclusiononly the insights.
```

B　追加分析：best-of-n 評価

3 回の試行のうち少なくとも 1 回成功した場合を成功とみなした best-of-n 評価を行った結果を図 4 に示す．ヒントなし条件との差が小さくなったことや，抽象手順-低関連度の性能が大幅に向上したことから，単一試行に基づく評価ではヒントの潜在的な有効性を十分に捉えられない可能性が示唆される．

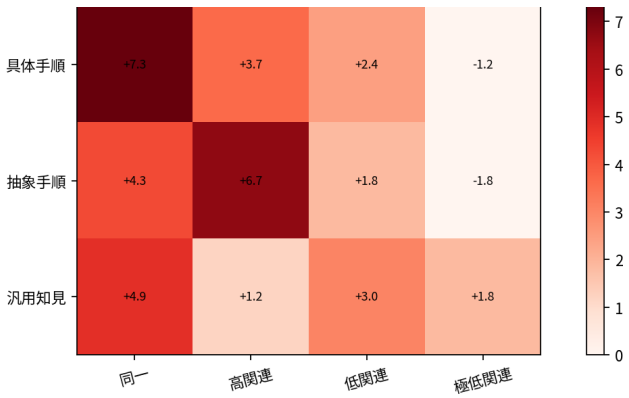

図 4: best-of-n（n=3）評価における成功率のヒントなし条件との差分．