大規模言語モデルが持つ抽象推論能力の分析

清野輝風 [1]　青木洋一 [1,2]　斉藤いつみ [1,2]　坂口慶祐 [1,2]
[1] 東北大学　[2] 理化学研究所
seino.terukaze.p8@dc.tohoku.ac.jp　youichi.aoki.p2@dc.tohoku.ac.jp
itsumi.saito@tohoku.ac.jp　keisuke.sakaguchi@tohoku.ac.jp

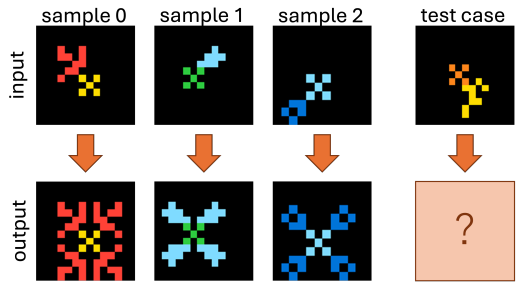
概要

人工知能が高度な汎化能力を持つためには，少数の例からパターンを抽出し，新しい入力に適用する抽象推論能力の向上が重要である．しかしながら，既存の大規模言語モデルは未だ十分な抽象推論能力を有しておらず，その能力について詳細な分析が必要である．本稿ではこれらの能力を評価する代表的なデータセットである ARC（Abstraction and Reasoning Corpus）について few-shot 学習を通して実際に解かせ，ARC の各タスクに必要な処理を能力ごとに分解し分類した上で，能力ごとの精度を分析することで大規模言語モデルの抽象推論能力を詳細に評価する．

1　はじめに

帰納推論能力とは，少数の例から共通するパターンを抽出し，新しいケースに適用させる能力である [1]．この能力は，事前学習データには存在しないような新しいパターンのタスクに遭遇した時でも，新たに学習データを用意するなどといった追加コストを支払うことなく頑健に対処するために必要である [2]．特に，抽象推論能力と呼ばれる図形やパズルなどといった記号を対象にした帰納推論能力は，学習コーパスの量が少ないため未見のパターンに対する適用能力の評価に適している [3]．しかしながら，既存の大規模言語モデルは未だ抽象推論能力が欠けており [1]，抽象推論能力について詳細な分析が必要である．

抽象推論能力を測定するための代表的なデータセットとして ARC（Abstraction and Reasoning Corpus）が挙げられる [2]．ARC のタスクの例を図 1 に示す．ARC は人間らしい汎用的かつ流動的な知能を持っているか評価するためのベンチマークである．具体的には，複数の入出力画像例から共通する操作を推論し，テストケースに当てはめる能力を調べるデータセットである．ARC を解くにあたって，画像や画像内の図形の認識および移動や回転などの操作，論理演算や算術演算などの能力が要求されている．ARC は数個の例から複雑なパターンを推論するという特徴から人間には解けるが人工知能には解けない傾向がある [2]．この問題によって ARC への注目が集まっており，コンペティションが開かれるほどである．現在，最も精度の高い手法を用いて約 50% の精度を達成している[1)]．



tag: [“pattern_expansion”, “pattern_rotation”, “pattern_reflection”]

図 1　ARC のタスク例．下のリストはタスクに付与されたタグの集合 [4] を表す．

本研究ではタグを用いて ARC のタスクに含まれるさまざまな推論パターンを分類し，大規模言語モデルが得意な操作および不得意な操作について調査する．タグについてはセクション 3 で説明する．実験結果より，画像の回転や複製などといった画像全体に対する操作が得意であり，画像内の図形に対する操作が不得意であることが得られた．

2　関連研究

ARC 関連のデータセット　ARC 関連のデータセットとして ConceptARC [5] が挙げられる．ConceptARC は 16 個のカテゴリーで分類されており，それぞれについてそのカテゴリーに沿った能力のみで解けるよう簡易化されている．しかし，概念

1）https://arcprize.org/2024-results (閲覧日 2024/01/09)

レベルでのカテゴライズのため，具体的な操作に対する調査は難しく，また ARC のタスクに含まれているカテゴリを抽出しているため，オリジナルの ARC よりも表現空間は小さいものである．ConceptARC 以外にも Mini-ARC [6] というデータセットが存在する．このデータセットはグリッドサイズが 5 × 5 で一定になっており，6 つのカテゴリーで分類されている．グリッドサイズが固定であるにも関わらずデータセットの難易度としては ARC と非常に似ている [7]．他に，1D-ARC と呼ばれるデータセットが存在する．このデータセットはグリッドが 1 次元に固定されている．このデータセットは元の ARC に比べ大規模言語モデルの正答率が高い [8]．

大規模言語モデルでの ARC の解法　大規模言語モデルを用いて ARC を解くためにさまざまなアプローチが研究されている．例えば，入力をオブジェクトベースのグラフ構造で表現することで精度の向上を試みた研究がある [8]．他にも，ARC のタスクを解くコードを生成する手法と解答を直接生成する手法を組み合わせる研究 [9] や test-time training (TTT) を用いる研究がある [10]．さらに，GPT-4 と GPT-4V を比較してマルチモーダルモデルの推論能力を評価している研究もある [11]．加えて大規模言語モデルに仮説を複数生成させ，適切な仮説を選択してモデルにフィードバックをすることで推論を補強しようとする試みが行われている [12]．

大規模言語モデルの能力分析　大規模言語モデルの推論能力について特定の能力に焦点を当てた分析がいくつか行われている．例えば，大規模言語モデルが ARC に苦戦している理由として画像内のオブジェクトの認識能力が不足していると分析している論文がある [8]．ARC 以外にも，パターン認識能力と抽象推論能力を必要とするタスクにおいて，渦巻きの向きなどを含む基本的な概念が不足していると分析している論文もある [13]．

3　実験

カテゴリについて　本稿では能力のカテゴライズに Davide [4] により提案されているタグを用いる．タグは ARC の訓練データ全てに対して付与されており，図 1 のように各タスクに対して人間が解く際のいくつかの操作に対してそれぞれ対応したタグが付与されている．タグの種類は全部で 132 種類あり，グリッド画像全体やグリッド内の図形といった操作対象のオブジェクトや複製や反転などといった具体的な操作について区別されている．

表 1　全体の正答率 (1〜3-shot)

	正答率 (%)		
モデル名	1–shot	2–shot	3–shot
GPT-4o	7.46	9.50	13.7
Claude-3.5-sonnet	7.25	13.4	19.0
o1-preview	16.1	17.5	27.3

表 2　各タグの正答率上位 5 個 (3–shot)

	正答率 (%)		
タグ名	GPT-4o	Claude3.5	o1
image_reflection	44.4	44.4	66.7
image_expansion	45.5	40.9	45.5
image_rotation	28.6	42.9	57.1
associate_colors_to_colors	28.6	28.6	57.1
image_repetition	33.3	33.3	42.9

実験設定　本稿では，最先端のモデルである GPT-4o[2] [14], Claude3.5[3] [15], o1[4] [16] を用いて実験を行った．まず，ARC のトレーニングデータセットに対してサンプルを few-shot で見せた後，実際に解かせ全体の正答率を調査した．次に、正解したタスクに付与されていたタグ毎の正答率を調査した．その際，プロンプトについてユーザプロンプトとアシスタントプロンプトは 3 つのモデルで同じプロンプトを使用しマルチターン会話を形成した．プロンプトの詳細は付録 C を参照．また，定量的な結果を得るため，個数の少ないタグについてはフィルタリングを行い，個数の多い 50 種類のタグに対して分析した．

4　実験結果

4.1　定量的分析

全体の正答率　GPT-4o, Claude-3.5, o1 の 3 つのモデルの正答率を表 1 に示す．3 つのモデルの中で o1 が最も正答率が高く，3-shot の場合で約 27% であった．Claude-3.5 と GPT-4o については正答率がそれぞれ 3-shot で約 19%，約 14% であった．また，shot 数の増加とともに正答率が大きく上昇した．

タグごとの正答率　各タグに対する正答率を 3 つのモデルでの平均値ごとに並べたもののうち，上位 5 個と下位 5 個をそれぞれ表 2,3 に示す．全体の結果については付録 A に示す．表 2 から，最も正答率の高いタグは image_reflection（画像の表裏反転）であり，次に正答率の高いタグは

2）　gpt-4o-2024-08-06
3）　claude-3-5-sonnet-20240620
4）　o1-preview-2024-09-12

表 3　各タグの正答率下位 5 個 (3–shot)

	正答率 (%)		
タグ名	GPT-4o	Claude3.5	o1
pattern_resizing	0.00	0.00	12.5
pattern_reflection	0.00	6.67	0.00
pattern_rotation	0.00	0.00	0.00
bring_patterns_close	0.00	0.00	0.00
background_filling	0.00	0.00	0.00

表 4　o1 に対する各タグの正答率上位 5 個 (3–shot)

	正答率 (%)		
タグ名	GPT-4o	Claude3.5	o1
take_complement	0.00	0.00	71.4
image_reflection	44.4	44.4	66.7
summarize	17.6	17.6	58.8
image_rotation	28.6	42.9	57.1
associate_colors_to_colors	28.6	28.6	57.1

image_expansion（画像の拡張）であった．その次に高いタグは image_rotation（画像の回転）であった．これらのタグは画像全体に対する操作をするタスクに含まれており，このことから画像全体を対象にした操作が得意であることが得られた．

表 3 から，正答率の下位 5 個のタグに pattern_resizing（図形のサイズ変更）, pattern_reflection（図形の表裏反転）, pattern_rotation（図形の回転）が含まれていた．どのタグについても正答率は全体の正答率を大きく下回っており，pattern_rotation に関しては全てのモデルに対して正答率が 0 であった．これらのタグは全て画像内の局所的なパターンに対する操作のタスクに含まれており，このことから画像内の図形やパターンに対する操作が不得意であることが結果から得られた．

モデルごとの性能差について　o1 での正答率の上位 5 個のタグを表 4 に示す．全てのタグにおけるモデルの性能差は付録 A を参照．全体の傾向としては，ほとんどのタグにおいて o1 が最も正答率が高く，GPT-4o と Claude-3.5 では似た正答率の傾向が見られた．

表 4 より，o1 について最も正答率が高かったタグは take_complement であった．このタグは図 2 に示すような画像内に含まれる 2 つの小さな画像に対して同じ位置のピクセル同士で OR や AND などの論理演算を行うようなタスクに含まれており，このことからピクセル単位での OR 演算，AND 演算などの論理演算能力に長けていることが結果から得られた．また，GPT-4o や Claude-3.5 では take_complement の正答率が 0 であったことから，2 つのグリッドの各ピクセルに対して論理演算を行う能力を持ち合わせていないことが判明した．

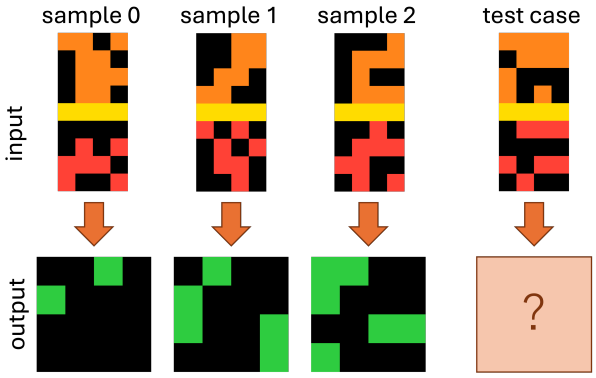


図 2　take_complement が付与されているタスク例

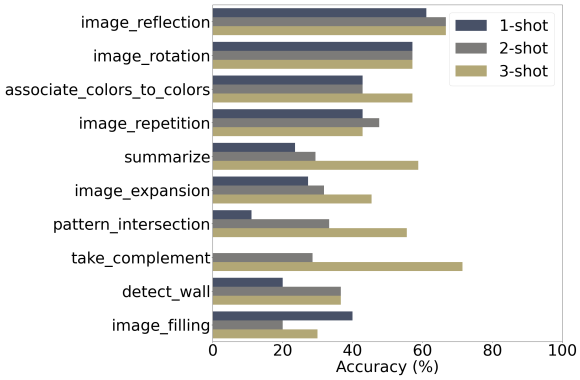


図 3　o1 で shot 数を変化させた時の各タグの正答率 (上位 10 個)

ショット数を変化させた場合　図 3 は，o1 で shot 数を変化させた時の各タグの正答率の変化を示したものである．結果から，image_reflection や image_rotation などのどのモデルでも正答率の高かったタグは 1-shot での正答率も高く，take_complement などといった，o1 のみ正答率の高かったタグは 1-shot での正答率が低く shot 数を増やすとともに正答率が増加していることが判明した．

このことから，o1 はグリッド全体に対する操作は 1 個の例で対応可能な簡単な問題であり，論理演算操作については全画像に共通する論理演算のパターンを推論することで精度を向上させていることが示唆される．

補足情報として画像を入れた場合　図 4 は，プロンプトに各例の入力を画像にしたものを補足情報として加えた場合の各タグの正答率の変化を示したものである．凡例について，接頭辞はモデル名を表し，接尾辞は画像の有無を表している．全体の傾向として，全てのケースで正答率が同じもしくは各モデルについて画像の有無関係なく正答率が同じタグ

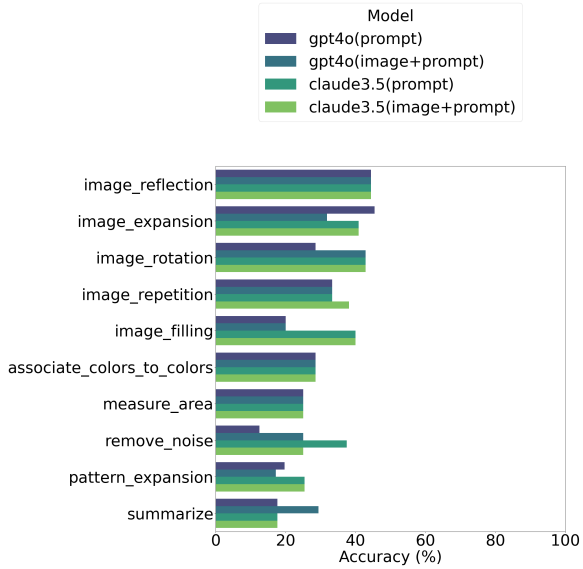


図 4　GPT-4o,Claude-3.5 について画像の有無による各タグの正答率 (上位 10 個)

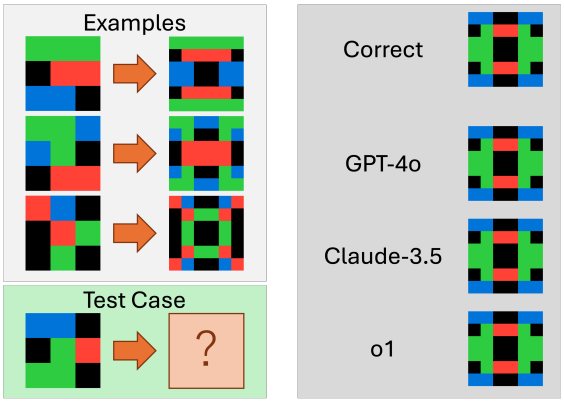
が散見された．また，画像を加えないケースの方が正答率の高いタグもあり，このことから画像が補足情報として機能していないことが示唆される．

4.2　定性的分析

4.1 から image_reflection の方が image_rotation よりも正答率が高いことが明らかとなり，これはグリッド全体に対して二次元的に回転させる操作よりも表裏反転させる操作の方が得意であることを示している．これらのタグはどちらも画像を回す操作であるという点で同系統であるにも関わらず正答率に差が見られ，2 つの操作の差について比較するためタスク単位での追加分析を行った．その結果，スコアが低下した原因は回転ではなく反転を行ったためであることが判明した．

図 5 はそれぞれ image_reflection のタグがついていた反転のタスクとそれに対するモデルの解答を可視化したものである．また，image_rotation のタグが付与された回転のタスクとそれに対するモデルの解答については付録 B を参照．分析から表裏反転のタスクについては全てのモデルで正答できているのに対し，回転のタスクについてはどのモデルでも全く回答できておらず，代わりに反転のような操作を行うような傾向が見られた．このことはモデルが回転系の操作を苦手としている可能性を示している．

一方，image_rotation が付けられているタスクの中でも，180 度回転させるタスクについては 3 つのうち 2 つのモデルで正答できている．このように 90 度回転させるタスクではどのモデルでも解けなかったのにも関わらず，180 度回転させるタスクについては正答できた理由に，反転でタスクを解いた可能性が挙げられる．180 度回転させた画像は元の画像を上下左右に 1 回ずつ表裏反転させる操作でも一致する．したがって，180 度回転させるタスクは反転操作を二回行って解いた可能性があり，人間とのタスク解決のプロセスの相違が示唆される．



(a) 反転のタスク例　(b) モデルの解答 (3-shot)

図 5　反転のタスク例とモデルの解答．

以上のことから，モデルは反転させることには長けているが回転させる操作は概念として保持しておらず，180 度回転タスクを反転の操作で解いたことを考慮するとスコア以上に回転させる能力が不得意である可能性が示唆される．

5　おわりに

本稿では，大規模言語モデルの抽象推論についてどのような能力が得意なのかあるいは不得意なのか，という問いに対して ARC のデータセットを用いて分析した．GPT-4o，Claude-3.5-sonnet，openAI o1-preview といった最先端のモデル 3 つに対して ARC を解かせ，タグによる分析を行ったところグリッド全体に対する操作が得意であることが確認できた．さらに細かく見ていくと，グリッド画像の表裏反転については得意であるが，グリッド画像の回転については操作として持ち合わせていない可能性が示された．

今後の展望として，個々の能力を獲得するためにデータを追加生成する場合どれくらいのリソースが必要なのか実験を行いたいと考えている．また，タグを補足情報として入れた際の ARC タスクの正答率の変化についても実験を行いたいと考えている．

謝辞

本研究は JST 次世代研究者挑戦的研究プログラム JPMJSP2114，JSPS 科研費 JP21K21343, JP22H00524 の助成を受けたものです．また，本研究を進めるにあたり，頻繁に議論に参加していただいた東北大学自然言語処理研究グループの皆様へ感謝いたします．

参考文献

[1] Gaël Gendron, Qiming Bao, Michael Witbrock, and Gillian Dobbie. Large language models are not strong abstract reasoners. **arXiv preprint arXiv:2305.19555**, 2023.

[2] François Chollet. On the measure of intelligence. **arXiv preprint arXiv:1911.01547**, 2019.

[3] Yile Wang, Sijie Cheng, Zixin Sun, Peng Li, and Yang Liu. Speak it out: Solving symbol-related problems with symbol-to-language conversion for language models. **arXiv preprint arXiv:2401.11725**, 2024.

[4] Davide Bonin. Task tagging, 2019. https://www.kaggle.com/code/davidbnn92/task-tagging.

[5] Arseny Moskvichev, Victor Vikram Odouard, and Melanie Mitchell. The conceptarc benchmark: Evaluating understanding and generalization in the arc domain. **arXiv preprint arXiv:2305.07141**, 2023.

[6] Subin Kim, Prin Phunyaphibarn, Donghyun Ahn, and Sundong Kim. Playgrounds for abstraction and reasoning. In **NeurIPS Workshop on neuro Causal and Symbolic AI**, 2022.

[7] Seungpil Lee, Woochang Sim, Donghyeon Shin, Wongyu Seo, Jiwon Park, Seokki Lee, Sanha Hwang, Sejin Kim, and Sundong Kim. Reasoning abilities of large language models: In-depth analysis on the abstraction and reasoning corpus. **arXiv preprint arXiv:2403.11793**, 2024.

[8] Yudong Xu, Wenhao Li, Pashootan Vaezipoor, Scott Sanner, and Elias B Khalil. Llms and the abstraction and reasoning corpus: Successes, failures, and the importance of object-based representations. **arXiv preprint arXiv:2305.18354**, 2023.

[9] Wen-Ding Li, Keya Hu, Carter Larsen, Yuqing Wu, Simon Alford, Caleb Woo, Spencer M Dunn, Hao Tang, Michelangelo Naim, Dat Nguyen, et al. Combining induction and transduction for abstract reasoning. **arXiv preprint arXiv:2411.02272**, 2024.

[10] Ekin Akyürek, Mehul Damani, Linlu Qiu, Han Guo, Yoon Kim, and Jacob Andreas. The surprising effectiveness of test-time training for abstract reasoning. **arXiv preprint arXiv:2411.07279**, 2024.

[11] Melanie Mitchell, Alessandro B Palmarini, and Arseny Moskvichev. Comparing humans, gpt-4, and gpt-4v on abstraction and reasoning tasks. **arXiv preprint arXiv:2311.09247**, 2023.

[12] Linlu Qiu, Liwei Jiang, Ximing Lu, Melanie Sclar, Valentina Pyatkin, Chandra Bhagavatula, Bailin Wang, Yoon Kim, Yejin Choi, Nouha Dziri, et al. Phenomenal yet puzzling: Testing inductive reasoning capabilities of language models with hypothesis refinement. **arXiv preprint arXiv:2310.08559**, 2023.

[13] Antonia Wüst, Tim Tobiasch, Lukas Helff, Devendra S Dhami, Constantin A Rothkopf, and Kristian Kersting. Bongard in wonderland: Visual puzzles that still make ai go mad? **arXiv preprint arXiv:2410.19546**, 2024.

[14] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. **arXiv preprint arXiv:2303.08774**, 2023.

[15] Anthropic. Claude 3 model card, 2024. https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf.

[16] OpenAI. Learning to reason with llms, 2024. https://openai.com/index/learning-to-reason-with-llms/.

[17] Guillermo Barbadillo. Few-shot prompting for arc24, 2024. https://www.kaggle.com/code/ironbar/few-shot-prompting-for-arc24.

A　各タグの正答率

本研究で対象にしたタグ全ての正答率を図 6 に示す．

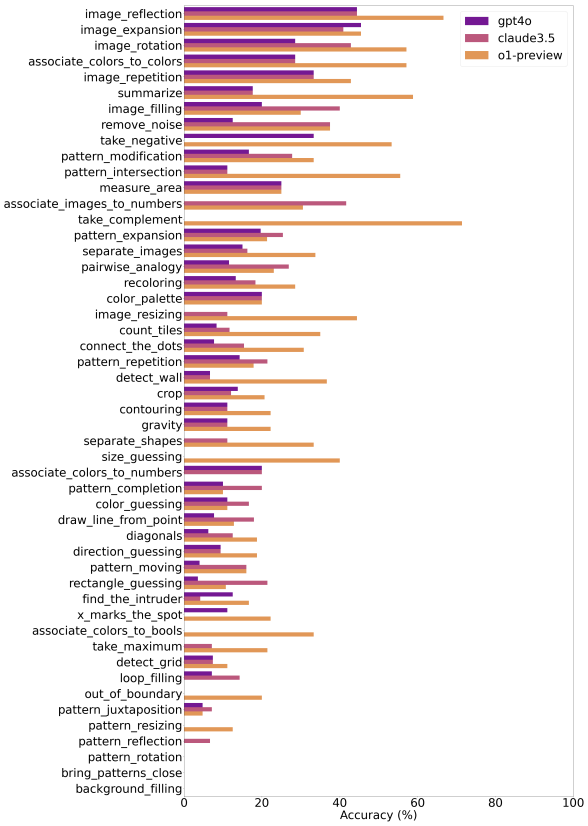
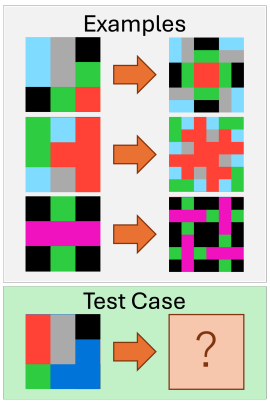


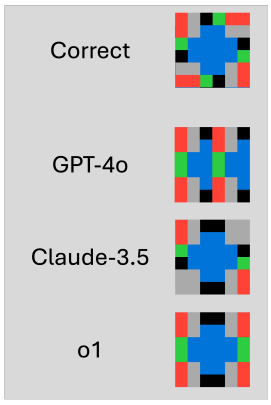
図 6　各タグの正答率 (3-shot)

B　回転のタスクとモデルの解答の可視化

image_rotation のタグが付いたタスクの例を図 7,8 に示す．

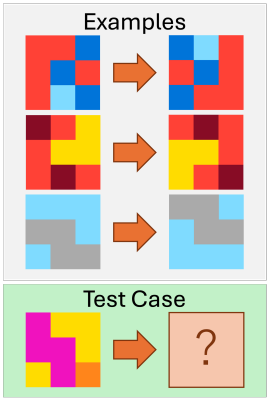


(a) 回転のタスク例



(b) モデルの解答 (3-shot)

図 7　回転のタスク例とモデルの解答．



(a) 180度回転タスク例

(b) モデルの解答 (3-shot)

図 8　180 度回転のタスク例とモデルの解答．

C　入力プロンプト

実験に用いた入力プロンプトを図 9 に示す．o1 モデルについてはシステムプロンプト（灰色の領域）は入れていない．プロンプト内の赤文字は画像を入れる際に追加した箇所である．プロンプトの作成にあたり Guillermo の kaggle の notebook [17] を参考にし，本研究用にプロンプトを変更した．

```
You are a helpful AI assistant. Your job is to solve tasks from the Abstraction and
Reasoning Challenge (ARC).
The user will present you with sample input and output grids for each task.
The puzzle-like inputs and outputs present a grid where each square can be one of ten
colors. A grid can be any height or width between 1x1 and 30x30.
The background of the grid is typically colored with 0.
The mapping between numbers and colors on the grid is as follows:

0 : black
1 : blue
2 : red
3 : green
4 : yellow
5 : gray
6 : magenta
7 : orange
8 : sky
9 : brown

The tasks from ARC are based on the following priors:

- Objectness: Objects persist and cannot appear or disappear without reason. Objects
can interact or not depending on the circumstances.
- Goal-directed: Objects can be animate or inanimate. Some objects are "agents" -
they have intentions and they pursue goals.
- Numbers & counting: Objects can be counted or sorted by their shape, appearance, or
movement using basic mathematics like addition, subtraction, and comparison.
- Basic geometry & topology: Objects can be shapes like rectangles, triangles, and
circles which can be mirrored, rotated, translated, deformed, combined, repeated, etc.
Differences in distances can be detected.

Image 1: <image>

Let's see if you can solve this simple ARC task. These are one input grid that define
the task. A grid can be any height or width between 1x1 and 30x30.
Now you should output the converted grids by recognizing the pattern from the
examples and applying the pattern to following test cases. Image 1 visualizes the
following input.

## Example 0
### Input
```grid
077
777
077
```

Output
```grid
000077077
000777777
000077077
077077077
777777777
077077077
000077077
000777777
000077077
```

Image 1: <image>

Let's see if you can solve this simple ARC task. These are one input grid that define
the task. A grid can be any height or width between 1x1 and 30x30.
Now you should output the converted grids by recognizing the pattern from the
examples and applying the pattern to following test cases. Image 1 visualizes the
following input.

Test case
Input
```grid
707
707
770
```
```

図 9　入力プロンプト例