いつ考え，いつ即答するか
文書理解視覚言語モデルにおける推論ルーティングの評価

Mengsay Loem[1]　橋本 航[1]
[1]Sansan 株式会社
{mengsay.loem,wataru.hashimoto}@sansan.com

概要

Chain-of-thought（CoT）は言語モデルにおける推論能力向上に有効とされる一方で，視覚言語モデルにおける適用効果は十分に整理されていない．本研究では，文書理解タスクを知覚中心タスクと推論中心タスクに分類し，推論の効果を統制した比較実験により CoT の有効性を検証する．視覚言語モデルを対象として，指示追従型モデルにおける CoT の有無，および推論型モデルにおける推論過程の長さ制約を比較する．その結果，指示追従型モデルでは，CoT が推論中心タスクの性能を大幅に改善する一方で，知覚中心タスクでは改善が小さいことを確認した．また，推論型モデルでは，推論過程を長くしても性能改善が限定的であることが一貫して観測された．さらに，必要な場合にのみ CoT を起動する軽量ルーティングの有効性を評価する．

1　序論

Chain-of-thought (CoT) プロンプティングは，中間の推論過程を明示的に生成させることで，言語モデルの推論性能を向上させる代表的手法である [1, 2]．近年は CoT をマルチモーダルの領域へ拡張し，視覚言語モデル (Vision Language Model; VLM) においても推論過程の明示が視覚推論ベンチマークで有効となり得ることが報告されている [3, 4]．しかし，既存研究の多くは自然画像の質問応答を中心とした設定を対象としている．そのため，複雑なレイアウト，微小文字の読み取り，厳密な出力形式を同時に要求する文書理解タスクにおいて，CoT がいつ有効で，なぜ効かない場合が生じるのかについては，十分に整理されていない．

VLM による文書理解は，帳票や図表を含む文書画像に対し，OCR パイプラインを介さずに読み取り・推論を行う用途として実利用が進んでいる [5, 6, 7, 8]．この設定において，(1) 複雑なレイアウト中の小さな文字を正確に認識・位置特定する微細な視覚知覚と，(2) 抽出した値を用いた集計・比較・算術などの視覚推論が強く結びつく．例えば，MMDocBench は，文書理解を対象として，タスクを知覚中心と推論中心の二群に整理したベンチマークとして提案された [9]．しかし，文書ベンチマーク上の既存評価は単一のプロンプト様式に固定されることが多く，明示的推論（CoT）の効果を，モデル規模やアーキテクチャ等の交絡要因から切り分けて理解することが難しい．

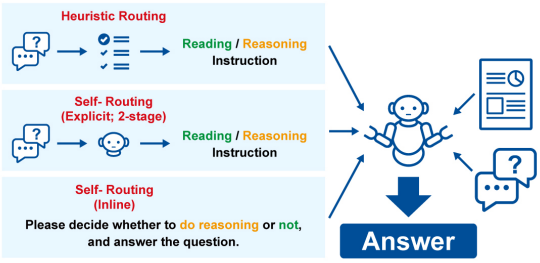


図 1: 推論ルーティングの概要．

本研究では，MMDocBench 上で VLM における明示的推論の効果を統制して比較する．複数の VLM を対象に，指示追従型 (Instruct 型) と，推論過程を出力する推論型 (Thinking 型) の二系統を評価する．Instruct 型については直接回答する Direct 回答と CoT の比較を行い，Thinking 型については推論過程の長さを Short/Long 制約で制御する．さらに，マルチモーダルジャッジモデル [10, 11] を用いて各出力を主要失敗要因と推論品質で注釈し，明示的推論がどの条件で機能するかを分析する．また，CoT の出力は，場合によっては誤った推論や形式逸脱を増やす可能性があるため，タスク・入力ごとに CoT を選択的に起動する軽量ルーティングを評価し，固定方針（常に Direct/CoT）に対する実用的な代替案を検討する．図 1 に，本研究で用いる推論ルーティングの概

要を示す．

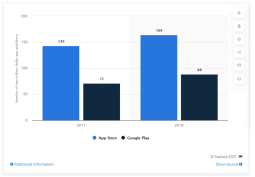
本研究の主な知見は次の 3 点である．

1. Instruct 型モデルでは，CoT は推論中心タスクで大きな改善をもたらす一方，知覚中心タスクでは改善が小さい．また，Thinking 型モデルでは，推論過程を長くしても改善は限定的である．
2. ジャッジモデルによる注釈から，知覚中心タスクの失敗は知覚と出力形式のボトルネックに支配され推論の有用性が低いのに対し，推論中心タスクの失敗は論理的ミスが中心で高品質かつ有用な推論が伴うことを示す．
3. 入力ごとに CoT を選択する単純なルーティングでも，知覚中心タスクを悪化させずに推論中心タスクの性能を改善でき，文書理解向け VLM の運用における適応的推論の有効性を示す．

2 実験設定

2.1 タスクとデータセット

本研究では，文書理解ベンチマークである MMDocBench [9] を用いる．MMDocBench は，帳票，科学図表等，多様な文書種を含み，計 15 種類のタスクタイプを定義し，Text Recognition, Key Information Extraction 等を含む知覚中心（perception-centric; P-tasks と Arithmetic Reasoning, Comparision 等を含む推論中心（reasoning-centric; R-tasks）の 2 つのカテゴリに分類される．各サンプルは，文書画像，質問文，および画像上の矩形を伴う正解アノテーションから構成される．各サンプルの質問として，タスク定義と出力要件を含む `question` フィールドを入力として用いる．



Task:
Chart Question Answering
Question:
How many app publishers were in Apple's App Store in 2017?
Answer: 143
Bounding Box: [226, 222, 250, 235]

図 2: MMDocBench タスクの例．

2.2 評価モデル

オープン VLM を対象に，以下の 2 系統に大別したモデルを対象とする．

- **Instruct 型** 自然言語の指示に従う対話形式モデル: Qwen3-VL Instruct[1)2)] [12], InternVL[3)] [13], Gemma-3[4)] [14]
- **Thinking 型** 推論過程[5)]を生成するよう学習されたモデル: DeepSeek-VL2[6)] [15], Qwen3-VL-Thinking[7)8)] [12], GLM-4.1V[9)] [16]

2.3 プロンプト条件

本研究では，明示的推論を誘発する条件を統制して比較する．Instruct モデルには次の 2 条件を設定する．**Direct**: 質問と画像を入力し，説明なしで直接回答するよう指示する．**CoT**: 最終回答の前に段階的推論を行うよう短い指示（例: まず段階的に考え，その後に最終回答のみを出力せよ）を付与する．モデル間で出力抽出を安定化するため，最終回答は必ず新しい行に`<answer></answer>`として出力する制約を課す[10)]．Thinking モデルでは次の 2 設定を評価する．**Short**: 推論を簡潔に（例: `<think>`内を 1–2 文以内）保ち，必要最小限の根拠提示に留めるよう指示する．**Long**: 詳細な多段推論を`<think>`内で記述するよう促す．用いられるプロンプトの詳細は付録 A に記載する．

2.4 評価指標

モデルの最終回答と正解データの完全一致に基づく Exact Match (EM) を用いる．予測回答と正解回答に対して，空白・句読点の正規化等の軽量な正規化を施し，正解候補のいずれかと完全一致した割合を EM として算出する．EM は，全タスク平均に加えて，P-tasks および R-tasks それぞれで集計する[11)]．

3 推論は文書 VLM に有効か？

文書理解における明示的推論（CoT や推論過程）が，P-tasks と R-tasks で同程度に有効かを検証する．具体的に，Instruct 型モデルでは Direct と CoT を比較し，Thinking 型モデルでは Short と Long を比較する．表 1 に，各モデルの P-tasks/R-tasks における平

1） https://huggingface.co/Qwen/Qwen3-VL-4B-Instruct
2） https://huggingface.co/Qwen/Qwen3-VL-8B-Instruct
3） https://huggingface.co/OpenGVLab/InternVL2-4B
4） https://huggingface.co/google/gemma-3-4b-it
5） `<think></think>`などのタグ内に推論を出力する．
6） https://huggingface.co/deepseek-ai/deepseek-vl2-tiny
7） https://huggingface.co/Qwen/Qwen3-VL-4B-Thinking
8） https://huggingface.co/Qwen/Qwen3-VL-8B-Thinking
9） https://huggingface.co/zai-org/GLM-4.1V-9B-Thinking
10） モデルが`<answer>`タグを出力しない場合，最終行抽出等の単純なフォールバック規則で予測文字列を取得する．
11） 本研究の EM 評価では矩形の一致は評価対象としない．

表 1: Instruct 型と Thinking 型 VLM における，P-tasks および R-tasks の性能比較.

	モデル	知覚中心タスク (P-tasks)		推論中心タスク (R-tasks)	
		Direct/Short	CoT/Long	Direct/Short	CoT/Long
Instruct 型	gemma-3-4b-it	0.225	**0.236**	0.250	**0.310**
	InternVL2-4B	0.325	0.282	0.218	**0.250**
	Qwen3-VL-2B-Instruct	0.394	0.385	0.456	**0.544**
	Qwen3-VL-4B-Instruct	0.432	**0.452**	0.519	**0.650**
	Qwen3-VL-8B-Instruct	0.453	0.453	0.490	**0.547**
Thinking 型	deepseek-vl2-tiny	0.203	0.195	0.085	**0.089**
	Qwen3-VL-4B-Thinking	0.391	0.374	0.615	**0.629**
	Qwen3-VL-8B-Thinking	0.421	**0.423**	0.662	**0.690**
	GLM-4.1V-9B-Thinking	0.257	**0.277**	0.344	**0.357**

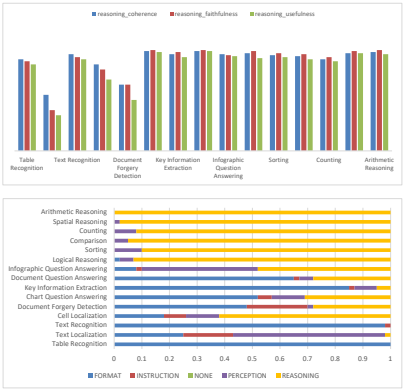


図 3: 推論の主要失敗要因と品質の自動評価.

均 EM を示す.

Instruct 型: CoT の利得は R-tasks に集中する Instruct 型モデルでは，CoT は R-tasks に対して一貫して大きな改善をもたらす一方，P-tasks では改善が小さいか不安定であり，場合によっては低下する．例えば Qwen3-VL-4B-Instruct では，R-tasks が Direct から CoT で大きく向上する一方で，P-tasks は小幅な変化に留まる．この傾向は他の Instruct モデルでも概ね共通であり，明示的推論は「比較・集計・算術」などの多段操作が主要なボトルネックとなる場面で特に有効であることを示唆する．

Thinking 型: 推論過程の長文化は精度向上に結びつきにくい Thinking 型モデルでは，Short から Long へ推論過程を長くしても，P-tasks/R-tasks のいずれにおいても精度差は小さく，明確な単調増加は観測されない．例えば Qwen3-VL-8B-Thinking では，R-tasks で Long が Short をわずかに上回るが，改善幅は限定的であり，P-tasks はほぼ同等である．

この非対称性は，P-tasks では知覚（読み取り・位置特定）や出力形式が支配的なボトルネックであり，推論の追加が介入しにくい可能性を示唆する．次節では，強力なマルチモーダルジャッジモデルを用いて，失敗要因 (Vision/Format/Instruction/Reasoning) の分布，および推論品質（整合性・忠実性・有用性）を分析し，この非対称性がどのような誤り構造から生じるのかを検証する．

4 推論はいつ忠実で有用か？

各サンプルに対し，Qwen3-VL-32B-Instruct [12] [12] をジャッジとして用い，文書画像，質問 (question)，正解テキスト，モデルの出力（<think>を含む）を入力する．ジャッジは (1) 主要失敗要因を Vision/Format/Instruction/Reasoning のいずれかで付与し，(2) 推論品質として Coherence（整合性），Faithfulness（画像への忠実性），Usefulness（解答への有用性）を [0, 1] で採点する (付録 A.4)．図 3 に Qwen3-VL-8B-Thinking の場合の集計を示す [13]．

P-tasks: 推論の Usefulness が低く不安定で，失敗は知覚と形式が支配的． P-tasks では，推論品質が全体として低く，特に Usefulness が不安定である（図 3 上部）．これは，認識・位置特定が主となるタスク (例：文字読み取りや位置特定) において顕著であり，モデルが「読むべき箇所」や「必要な処理」を一見整合的に述べていても，画像上の微小文字を誤読・見落としした時点で推論が解答に寄与しにくいことを示唆する．

この傾向は失敗要因の内訳からも裏付けられる．P-tasks の誤答は主として Vision（誤読・見落とし・誤定位）と Format(出力指示違反，タグや JSON 不整合，正規化失敗) により説明され，Reasoning が主要因となる割合は小さい (図 3 下部)．したがって P-tasks では，「見えていない・正しく書けない」

12） https://huggingface.co/Qwen/Qwen3-VL-32B-Instruct
13） Instruct 型モデルの場合の結果を付録 B に示す.

表 2: 推論ルーティングの性能比較．SR-E と SR-I は Self-Routing (Explicit) と Self-Routing (Inline) を表す．

タスク	Gemma-3-4B					InternVL2-4B					Qwen3-VL-4B-Instruct					Qwen3-VL-8B-Instruct				
	Direct	CoT	Heuristic	SR-E	SR-I	Direct	CoT	Heuristic	SR-E	SR-I	Direct	CoT	Heuristic	SR-E	SR-I	Direct	CoT	Heuristic	SR-E	SR-I
P-tasks	0.225	0.236	**0.250**	**0.245**	**0.247**	0.325	0.282	0.297	0.303	0.316	0.432	0.452	0.451	**0.461**	**0.462**	0.453	0.453	0.446	0.451	0.453
R-tasks	0.250	0.310	0.291	**0.316**	0.290	0.218	0.232	0.234	**0.243**	**0.234**	0.519	0.650	0.622	**0.704**	0.618	0.552	0.704	**0.713**	**0.711**	**0.718**

というボトルネックに対し，推論過程の追加は有効な介入になりにくい．また，軽い統合を含む P-tasks (Key Information Extraction や一部の QA 系) では Instruction/Reasoning 要因も混入しており，推論過程が不要な推測や形式逸脱を誘発するケースがあることも示唆される．

R-tasks: 推論品質は一貫して高い　R-tasks では，Coherence/Faithfulness/Usefulness が概ね高水準で安定している．誤答時の主要因はほぼ Reasoning に集中し，比較対象の取り違え，集計や算術の誤り，条件解釈ミスといった「操作の誤り」が中心となる．すなわち，R-tasks では，関連情報が知覚的に取得できている前提で，最終解に至る操作手順が性能を規定しており，段階的推論 (CoT) が精度に直結しやすい構造になっている．これは前節で観測された，Instruct 型モデルで CoT が R-tasks で大幅に改善という結果と整合的である．

Short vs. Long: 逓減効果と副作用　Thinking 型モデルにおいて，Short から Long へ推論トレースを冗長化しても，R-tasks では推論品質が既に高水準のため改善余地が小さく，精度利得は限定的である．一方 P-tasks では，Long 化しても Vision 起因のボトルネックは解消されにくく，出力が長くなることにより Format や Instruction 由来の事故が増えるケースがあり，推論が主要因であるか否かによって，推論過程の有効性が決まることが分かる．

5　推論を選択的に起動できるか？

入力ごとに CoT の有無を切り替えることで，固定方針（常に Direct/CoT）よりも良い性能が得られるかを検証する．前節の結果より，P-tasks では推論の有用性が低く，R-tasks では有用性が高いことが示唆されたため，必要な場合にのみ CoT を起動するルーティングは合理的な設計である．

設定　ルーティング実験は Instruct 型モデルに限定して行う[14]．各入力に対し，ルータが Direct または CoT のいずれかを選択し，選択されたプロンプトで最終回答を生成する．

ルーティング方法　軽量で実装可能性の高い 3 種の方策を比較する．(1) **Heuristic**: タスク種別と質問文の表層手掛かり（数値の出現，計算・比較を示す語等）に基づきルールベースで CoT 起動を判定する．(2) **Self-routing（Explicit）**：同一モデルに対して小さな一次判定（Direct/CoT の二値分類）を行い，その結果に従って本推論を実行する．(3) **Self-routing（Inline）**：一回の生成の中で，モデル自身に Direct/CoT を選択させ，選択をタグで明示させた上で回答させる．詳細は付録 A.3 に掲載する．

結果　表 2 に P-tasks/R-tasks に分けた平均 EM を示す．P-tasks では，いずれのモデルにおいてもルーティングの差は小さく，最良の固定方針（常に Direct/常に CoT）と同程度の EM に留まる．これは前節で示したように，P-tasks の主要失敗要因が Vision/Format であり，「考えるかどうか」を変えても根本的なボトルネックが変わりにくいことと整合的である．一方 R-tasks では，複数のモデルにおいてルーティングが常に CoT を概ね維持，あるいは上回る結果が観測される．例えば Qwen3-VL-4B-Instruct では，Heuristic が常に CoT を上回るスコアを示し，Qwen3-VL-8B-Instruct では Self-routing 系が高い R-task 性能を達成する．最良の方策はモデルによって異なるものの，R-tasks では CoT による改善を回収しつつ，P-tasks では悪化させにくい，という傾向は一貫している．

6　結論

文書理解タスクにおいて，明示的推論の効果を統制的に検証した．その結果，推論中心タスクでは CoT が Instruct 型 VLM を大きく改善する一方，知覚中心タスクでは効果が小さく，Thinking 型 VLM でも推論過程を長くしても改善は限定的であった．分析から，知覚中心タスクの失敗は視覚・形式要因が支配的で推論の有用性が低い一方，推論中心タスクでは論理要因が支配的で推論が有用であることが示唆された．以上から，文書 VLM における推論は万能ではなくボトルネックに応じて使い分けるべく，必要時のみ推論を発火するルーティングは実用的な設計として有望である．

14）前節において Thinking 型モデルは，推論過程を長くしても改善が小さいためである

参考文献

[1] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In **Advances in Neural Information Processing Systems**, pp. 24824–24837, 2022.

[2] Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In **Advances in Neural Information Processing Systems**, pp. 22199–22213, 2022.

[3] Zhuosheng Zhang, Aston Zhang, Mu Li, hai zhao, George Karypis, and Alex Smola. Multimodal chain-of-thought reasoning in language models. **Transactions on Machine Learning Research**, 2024.

[4] Ruohong Zhang, Bowen Zhang, Yanghao Li, Haotian Zhang, Zhiqing Sun, Zhe Gan, Yinfei Yang, Ruoming Pang, and Yiming Yang. Improve vision language model chain-of-thought reasoning. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, **Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 1631–1662, Vienna, Austria, July 2025. Association for Computational Linguistics.

[5] Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Ocr-free document understanding transformer. In **Computer Vision – ECCV 2022**, p. 498–517, 2022.

[6] Minesh Mathew, Dimosthenis Karatzas, and C.V. Jawahar. Docvqa: A dataset for vqa on document images. In **Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)**, pp. 2200–2209, January 2021.

[7] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, **Findings of the Association for Computational Linguistics: ACL 2022**, pp. 2263–2279, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[8] Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and C.V. Jawahar. Infographicvqa. In **Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)**, pp. 1697–1706, January 2022.

[9] Fengbin Zhu, Ziyang Liu, Xiang Yao Ng, Haohui Wu, Wenjie Wang, Fuli Feng, Chao Wang, Huanbo Luan, and Tat Seng Chua. Mmdocbench: Benchmarking large vision-language models for fine-grained visual document understanding, 2024.

[10] Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: NLG evaluation using gpt-4 with better human alignment. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, **Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing**, pp. 2511–2522, Singapore, December 2023. Association for Computational Linguistics.

[11] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In **Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track**, 2023.

[12] Shuai Bai, Yuxuan Cai, Ruizhe Chen, and et al. Qwen3-vl technical report, 2025.

[13] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**, pp. 24185–24198, 2024.

[14] Gemma Team and et al. Gemma 3 technical report, 2025.

[15] Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, Zhenda Xie, Yu Wu, Kai Hu, Jiawei Wang, Yaofeng Sun, Yukun Li, Yishi Piao, Kang Guan, Aixin Liu, Xin Xie, Yuxiang You, Kai Dong, Xingkai Yu, Haowei Zhang, Liang Zhao, Yisong Wang, and Chong Ruan. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding, 2024.

[16] V Team and et al. Glm-4.5v and glm-4.1v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning, 2026.

A　指示プロンプト

```
You are an expert document-understanding assistant.
- Always end with ONE final line:
      <answer>FINAL_ANSWER_HERE</answer>
- Do not output anything after </answer>.
Output format
- If you include visible reasoning, enclose it ONLY inside
      <think> ... </think>.
- On a NEW LINE, output ONLY:
      <answer>FINAL_ANSWER_HERE</answer>
- Do not print anything after </answer>.
```

A.1　指示追従型モデル (Instruct 型)

```
# Direct
System:
- Answer directly without any visible reasoning.
- Ignore any request for JSON/bbox; do not output it.
User:
Answer directly with no extra text.

# CoT
System:
- First, write your reasoning ONLY inside <think> ...
      </think>.
- Then provide the final answer.
User:
First, think step by step inside <think> ... </think>, then
      finalize.

# Self-Routing (Inline)
System:
- Decide if the question is simple (read/copy) or complex
      (multi-step).
- Output exactly one line before the final answer:
<mode>direct</mode> or <mode>cot</mode>
- If direct: no <think>. If cot: <think> only.
User:
Decide simple vs. complex; follow the corresponding behavior
      and emit <mode>.
```

A.2　推論型モデル (Thinking 型)

```
# Short
System:
- Answer directly; avoid visible reasoning.
- If absolutely necessary, ONE short sentence inside <think>
      ... </think>.
- Ignore any request for JSON/bbox; do not output it.
User:\\
If you must show a hint, keep it to ONE short sentence inside
      <think> ... </think>.

# Long
System:
- Provide DETAILED step-by-step reasoning ONLY inside <think>
      ... </think>.
User:
Reason carefully in multiple steps inside <think> ...
      </think>, then finalize.

# Self-Routing (Inline)
System:
- Choose one: no thinking short thinking / long thinking.
- Output exactly one line before the final answer:
<mode>no thinking</mode> or <mode>short thinking</mode> or
      <mode>long thinking</mode>
No thinking: no <think>. Short: ~2 sentences. Long: detailed
      steps.
\textbf{User:} Decide among no/short/long; follow it and emit
      <mode>.
```

A.3　ルーター用プロンプト

Self-Routing (Explicit) では，質問文のみの分類を実行し，Direct か CoT の判定を実施する．

```
System:
You are an expert at deciding whether a document question
requires step-by-step reasoning or simple reading.
Output ONLY one word: direct or cot.
Do not output anything else.

User:
Task: [[TASK]]
Sub-task: [[SUBTASK]]
Question: [[QUESTION]]
Decide whether to answer with Reading or Thinking.
```

Heuristic routing では，以下のいずれかの条件を満たす場合，CoT を選択する: (1) タスクが推論中心であることが既知の場合 (例: カウント，空間推論), (2) 質問に明示的な推論の手がかり (例: **difference**, **average**，**total**) が含まれている場合，または (3) 質問に複数の数値が含まれ，かつ長さが所定の閾値を超えている場合．それ以外の場合は，Direct をデフォルトとする．

A.4　ジャッジモデルへの指示プロンプト

```
System:
You are an expert evaluator for document understanding.
Given an image, a question, the ground-truth answer,
and a model response, evaluate correctness and reasoning
      quality (0-1).
Return your judgment in JSON format.

User:
Image: [[IMAGE]]
Question: [[QUESTION]]
Ground-truth answer: [[GROUND TRUTH]]
Model output: [[MODEL OUTPUT]]
Evaluate the model output and return a JSON object with:
primary error type: Vision, Format, Instruction, Reasoning.
reasoning coherence, faithfulness, usefulness.
```

B　Instruct 型モデルの推論の失敗要因と品質の評価

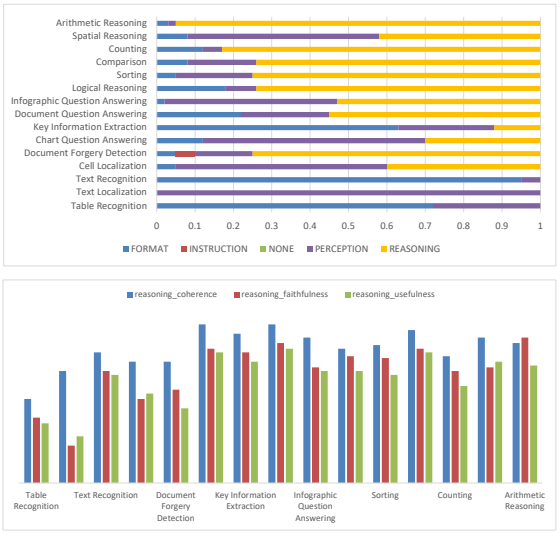


図 4: gemma3-4b の推論の主要失敗要因と品質の自動評価．