

LLM についての Issue 解決の難しさを 特徴量としたモデルルーティング

オープンソースソフトウェア工学研究室
S2520048 陸井優希

背景

大規模言語モデルの利活用が一般化

- モデルの数も増加し続けており種類も豊富
オープンソースモデル共有プラットフォームで公開

Models 2,863,854

2026/05/10時点のモデル登録数

モデルの人気度などを参考に利用するモデルを選択

SE分野におけるLLM利用

コード生成



2つの整数を受け取り，合計を返す関数を書け



フィボナッチ数列のうち素数でもある
n番目の値を返す関数を書け

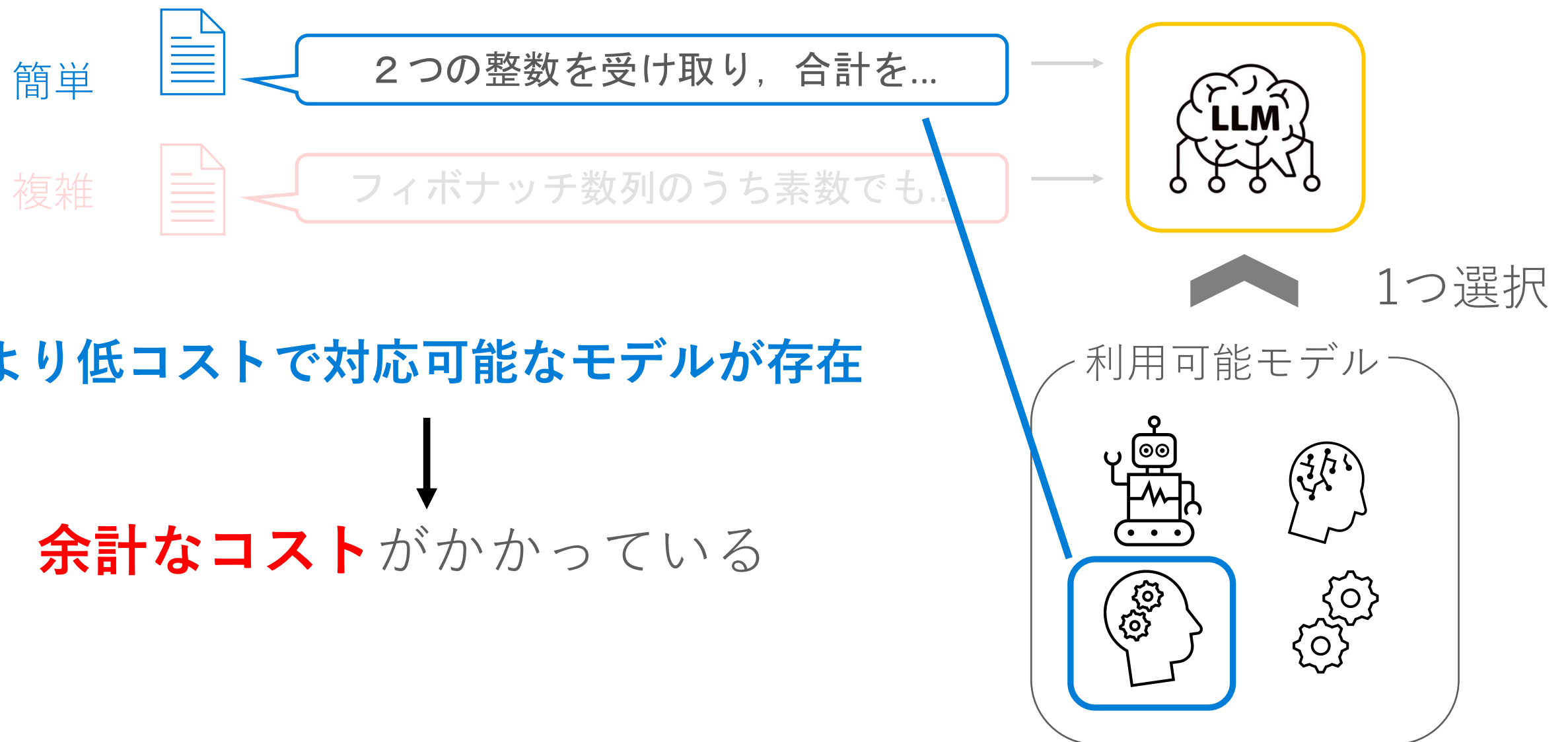
入力に関わらず一つのモデルで処理

SEタスク全般で同じ傾向^{[1][2][3][4]}

単一モデル利用の問題点

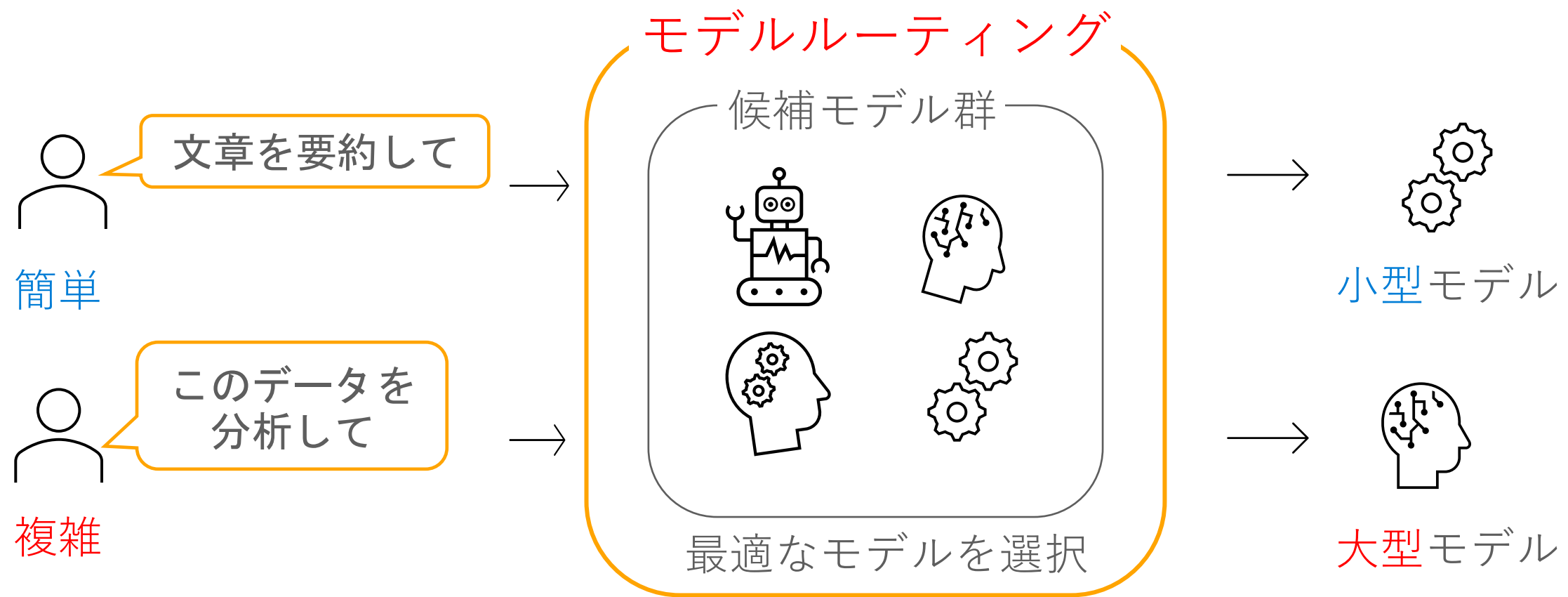
LLM利用にかかるコスト

- 高性能なモデルほど高価である傾向



コスト削減方法

クエリごとに最適なモデルを自動的に選択する技術



クエリごとに最適なモデルを選択することでコスト削減

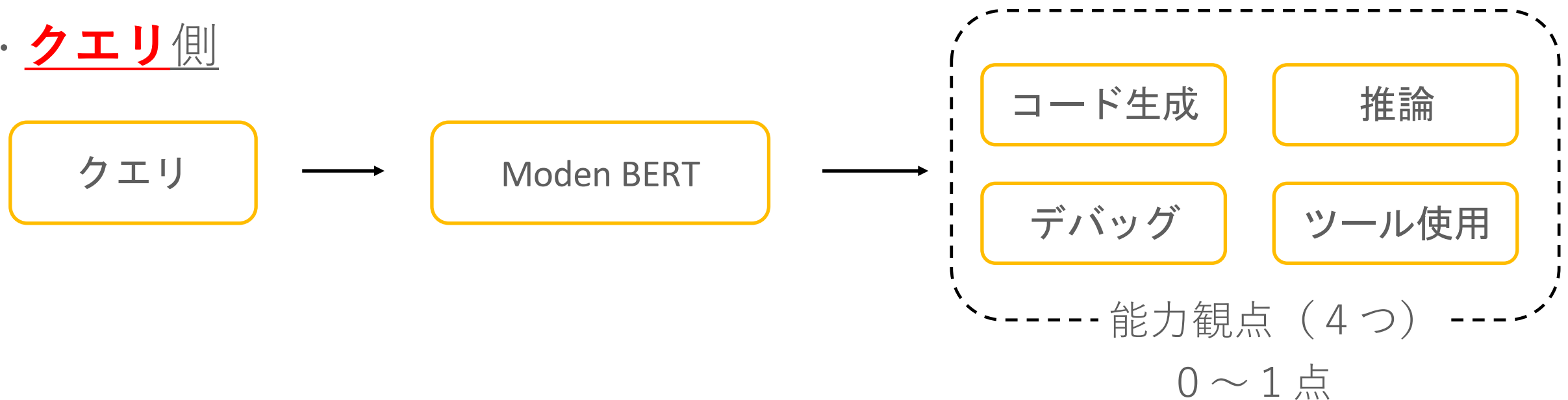
NLPタスクで複数の手法が提案・検証^{[5][6]}

SE分野でのモデルルーティング

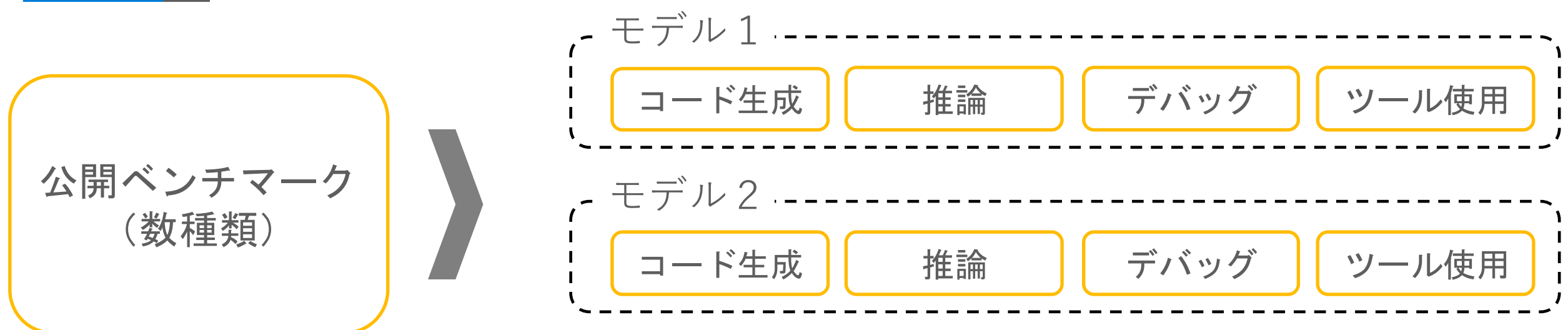
先行研究

- クエリが要求する能力の推定

・ クエリ側



・ モデル側



SE分野でのモデルルーティング

先行研究

- クエリが要求する能力の推定

クエリが要求する観点を満たす最も安いモデルを選択

コード生成

推論

デバッグ

ツール使用

能力観点（4つ）

0～1点

・ モデル側

公開ベンチマーク
(数種類)



モデル 1

コード生成

推論

デバッグ

ツール使用

モデル 2

コード生成

推論

デバッグ

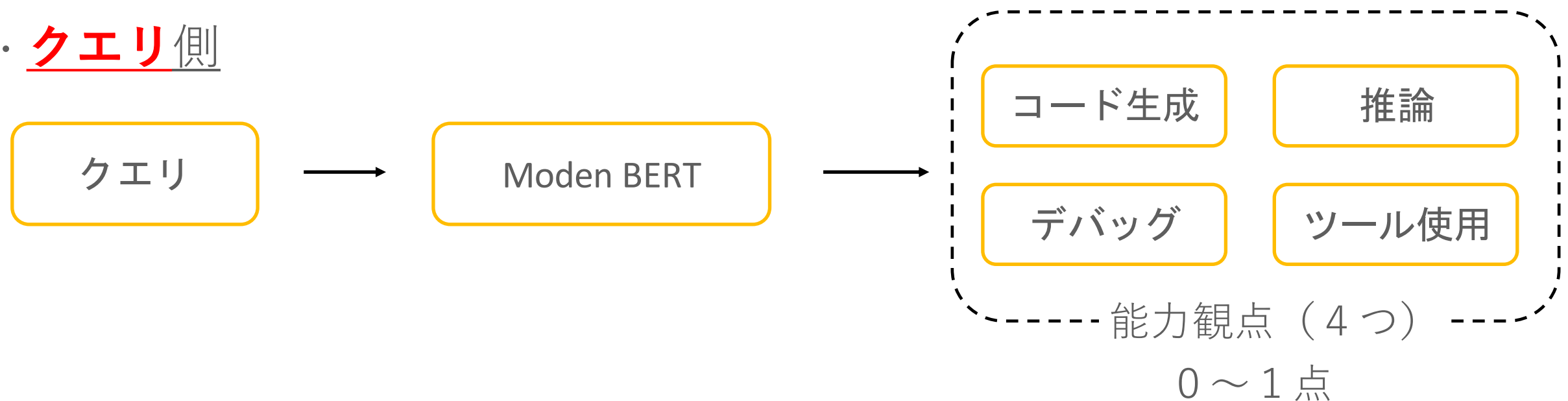
ツール使用

先行研究の問題点

先行研究

- クエリが要求する能力の推定

・ クエリ側



問題点

- LLM にとっての難しさがどこに現れるかを調べた研究を考慮していない

LLMにとっての難しさ

Issue本文から^[9]

修正案の有無 コードの引用 再現スクリプト 修正対象の明示
読みにくい散文 報告の短さ

リポジトリから^[10]

ファイル数 総サイズ 平均サイズ ディレクトリ数
最大の深さ 平均の深さ など

既存のルーティング手法はこれらの情報を利用していない

目的・アプローチ

目的

- LLMにとっての難しさを用いたルーティングの提案

アプローチ

- ルータを構築

調査で発見した特徴量を入力としたルータを構築

- 単一モデル利用と比較

単一のモデルを使い続けた場合の精度・コストと比較

ルータの構築

各候補モデルが正解する確率を予測



設定した閾値より高いモデルのうち安いモデルを選択 (例 閾値 = 0.7, 価格順: A > B > C)



評価実験

データセット

- SWE-bench Verified

公開されている回答記録を利用（11モデル）

評価指標

- 品質保持率（QR）

$$QR = \frac{\text{ルータの解決数}}{\text{理想ルータの解決数}}$$

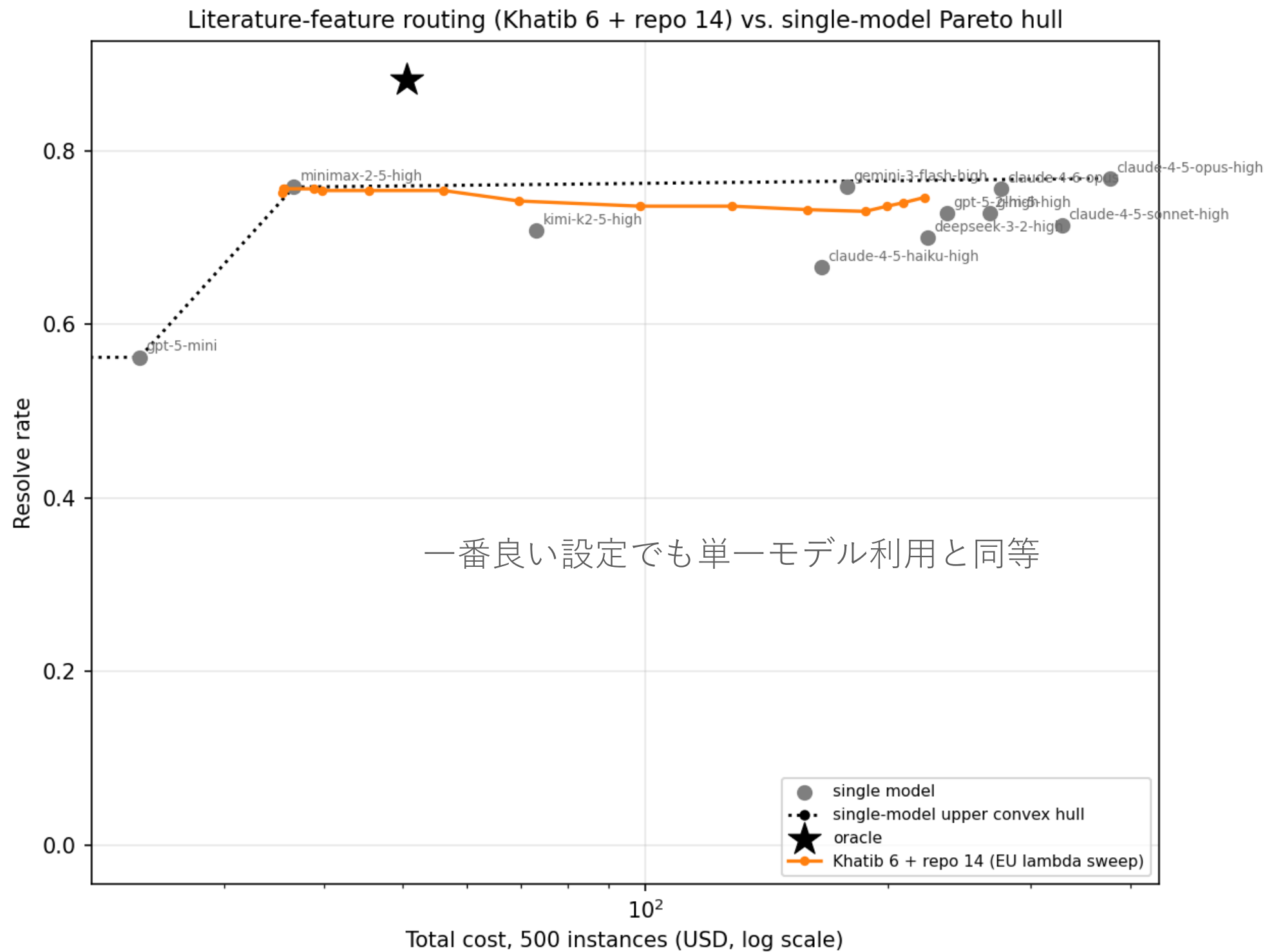
- コスト削減率（CS）

$$CS = 1 - \frac{\text{ルータのコスト}}{\text{最良モデルのコスト}}$$

- 誤ルーティング率

より安いモデルで解けたのに高いモデルを選んだ割合

結果



結果

手法	解決率(%)	コスト (\$)	QR	CS	Mis
理想ルータ	88.2	50.64	100	86.6	0
claude4.5 opus	76.8	376.95	87.1	0.0	86.8
MiniMax 2.5	75.8	36.64	85.9	90.3	33.0
提案	75.6	35.53	85.7	90.6	33.0

まとめ・今後の予定

AIについての難しさがどこに現れるかを文献から収集しルーティングに利用

単一モデル利用の場合と比較

単一モデル利用とほとんど同じ精度とコストに

今後の予定

- “ソフトウェア品質”を高めるようなルーティングを考える

参考文献

- [1] Dong et al., Self-collaboration code generation via ChatGPT, TOSEM 2024
- [2] Yuan et al., Evaluating and improving ChatGPT for unit test generation, FSE 2024
- [3] Xia et al., Automated program repair in the era of large pre-trained language models, ICSE 2023
- [4] Du et al., Vul-rag: Enhancing LLM-based vulnerability detection, 2024
- [5] MetaLLM: A High-performant and Cost-efficient Dynamic Framework for Wrapping LLMs Quang H. Nguyen et al., <https://arxiv.org/abs/2407.10834>
- [6] RouterDC: query-based router by dual contrastive learning for assembling large language models Chen, Shuhao et al., Proceedings of the 38th International Conference on Neural Information Processing Systems
- [7] IRT-Router: Effective and Interpretable Multi-LLM Routing via Item Response Theory Wei Song et al. ACL 2025 <https://aclanthology.org/2025.acl-long.761/>
- [8] BEST-Route: Adaptive LLM Routing with Test-Time Optimal Comput Dujian Ding et al. ICML 2025 <https://arxiv.org/abs/2506.22716>
- [9] Khatib et al., What Makes a Good Bug Report for an AI Agent?, arXiv 2026
- [10] Al-Haque and Johnson, What Makes Software Issue Resolution Tasks Difficult for Agents?, ESEM 2026