

生成AIのリスク低減に向けた AIモデルルーティングの活用

発表：山村慎太郎

共同研究 | 概要

Quality AI, 紀陽銀行, 和歌山大学の三者間で始まった, 共同研究プロジェクト

- Quality AI : 大学と協力した, オープンソースの活用
- 紀陽銀行 : 社内でAI活用・DX推進したい, 的なものの一環?
- 和歌山大学 : お金がほしい(本音)

GENIAC-PRIZE

- 経済産業省とNEDOが打ち出したプロジェクトの一環
 - 生成AIの社会実装を後押しする懸賞金活用型プログラム
- ひとまずこれを目標とし取り組む形に

共同研究 | GENIAC PRIZE

取り組む領域

生成AIの安全性確保に向けたリスク探索及びリスク低減技術の開発

生成AIの安全性に関わるリスクの特定と、当該リスク低減を目的とする技術を開発。妥当性や実効性、新規性等を審査。

懸賞金

第1位

7000万円

第2位

5000万円

第3位

3000万円

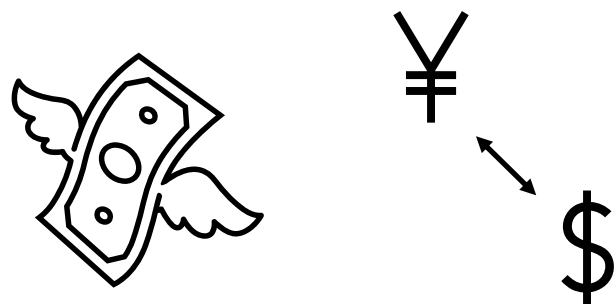
選んだ理由：研究テーマとの方向性の合致(と消去法)

- 次ページから、その研究内容の紹介

背景 | 社会・企業での生成AI活用

企業での商用LLM活用には様々な課題・制約が...

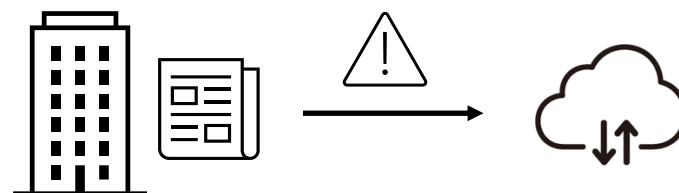
コスト問題



継続的にお金がかかる

- ・ 高額なAPI料金
- ・ 為替変動で支払額が予測困難に

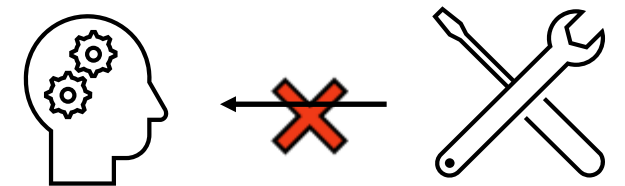
プライバシー問題



外部サービスへの
情報送信

- ・ 機密情報の取り扱い×
- ・ 社内データ漏洩の危険性

カスタマイズ



カスタマイズの限界

- ・ 自社特有タスクへの適応×
- ・ 専門知識の不足

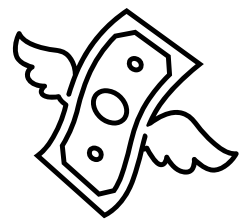
➤ 商用モデルでは、十分な活用に至れず

背景 | オープンソースLLM

モデルの重みや構造が無料公開されているLLM

- 例：Llama（Meta）， Mistral（Mistral AI） など
- 自社環境で運用可能で， 外部サーバに依存しない

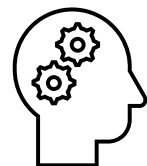
商用LLM



コスト問題



プライバシー問題



カスタマイズ制約



移行

オープンソースLLM

¥ 0

利用料金無料



プライバシー完備



カスタマイズ可

問題① | ハルシネーション

一般的に、商用モデルよりオープンソースモデルの方がハルシネーションが多い傾向

- Vectaraのhallucination-leaderboard_[1]より抜粋

商用モデル

Model	Hallucination Rate
Google Gemini-2.0-Flash-001	0.7 %
OpenAI o3-mini-high	0.8 %
Amazon Nova-Lite-V1	1.6 %
Anthropic Claude-Opus-4.1	4.2 %

オープンソースモデル

Model	Hallucination Rate
Qwen3-14B	2.2 %
DeepSeek-V2.5	2.4 %
Llama-3.3-70B-Instruct	4.0 %
Mistral-7B-Instruct-v0.3	9.5 %

経産省らの提示する「AIによるリスク」に該当_[2]

- “ハルシネーション等による誤った出力”という項目が明記

[1]<https://github.com/vectara/hallucination-leaderboard>

[2]「AI事業者ガイドライン（第1.1版）別添」（PDF形式：6,361KB）

問題② | オープンソースLLMの多様性

モデルの数・種類がとにかく多い

- Hugging Face
 - AIモデルの共有・活用のための世界最大級のプラットフォーム

Models 1,870,203

すでに膨大なモデル数

Parameters

< 1B 6B 12B 32B 128B > 500B

広範なモデルサイズ



Text Generation



Any-to-Any



Image-Text-to-Text



Image-to-Text



Image-to-Image



Text-to-Image



Text-to-Video



Text-to-Speech

+ 43

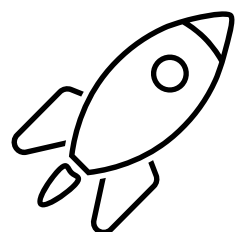
多様なタスクの種類

- さらに、数百～数千の新規モデルが毎日のように追加・更新

問題② | オープンソースLLMの多様性

モデルサイズと性能

大型LLM

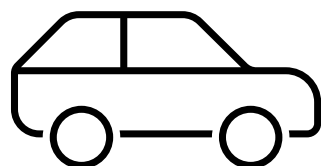


パワフルで
高性能



計算リソースが
大規模に

小型LLM



身軽だが
低性能



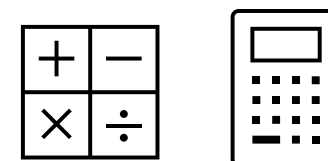
計算リソースは
小規模

モデルサイズと性能の
トレードオフの関係

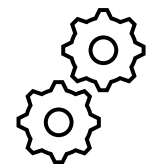
得意・不得意がある



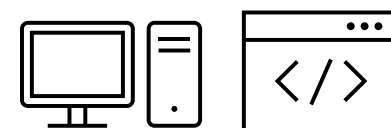
モデルA



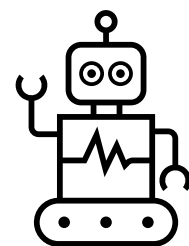
数学で高精度



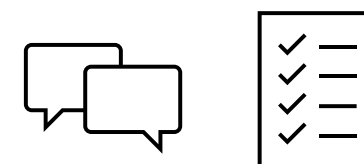
モデルB



コーディングで
高精度



モデルC



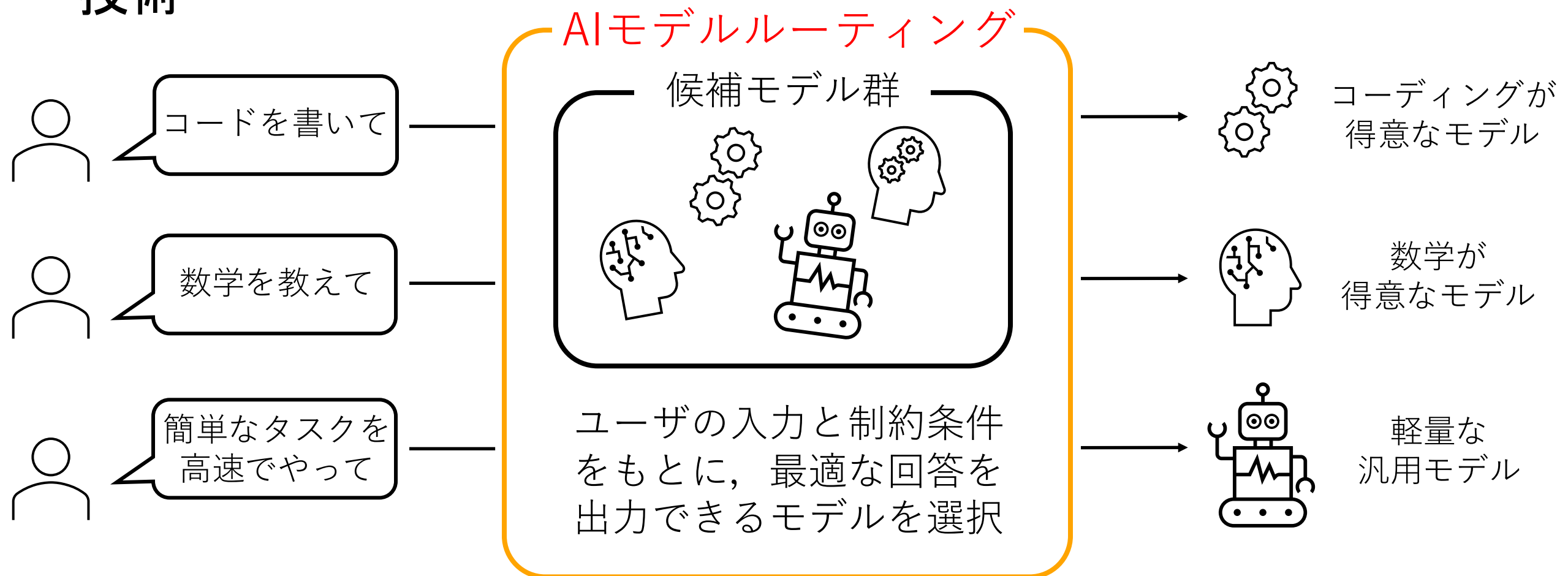
汎用タスクで
高精度

モデルごとに
最適なタスクが異なる

➤ 最適なモデル選択・評価が困難

AIモデルルーティング

タスク・クエリごとに、最適なAIモデルを自動的に選択する技術



➤ 常に最適なモデルを利用することで、誤回答の低減が可能？

- 社内完結、かつ高精度な出力の実現へ

目的 | 建前と本音

企業内でのオープンソースLLMの利活用による, 「AIによるリスク」の低減

- ローカルLLMによる社内完結で, 機密情報の流出リスクの低減
- AIモデルルーティングによる, ハルシネーションリスクの低減

ただし, 文脈によってはこの限りではない

- 既存研究で「AIリスクの低減」が語られることはほぼ無い
 - 特にハルシネーション
- 今回のテーマが「安全性・リスク低減」なので, そちらに文脈を寄せた形
 - “機密情報の流出”, “ハルシネーション等による誤った出力”はいずれも経産省が「AIによるリスク」として提示

アプローチ | 先行研究

“Universal Model Routing for Efficient LLM Inference”(Google, 2025)

Universal Model Routing for Efficient LLM Inference

Wittawat Jitkrittum Harikrishna Narasimhan Ankit Singh Rawat
wittawat@google.com hnarasimhan@google.com ankitsrawat@google.com

Jeevesh Juneja Congchao Wang Zifeng Wang
jeeveshjuneja@google.com congchaowang@google.com zifengw@google.com

Alec Go Chen-Yu Lee Pradeep Shenoy
ago@google.com chenylulee@google.com shenoypradeep@google.com

Rina Panigrahy Aditya Krishna Menon Sanjiv Kumar
rinap@google.com adityakmenon@google.com sanjivk@google.com

Google

Abstract

Model routing is a simple technique for reducing the inference cost of large language models (LLMs), wherein one maintains a pool of candidate LLMs, and learns to route each prompt to the smallest feasible LLM. Existing works focus on learning a router for a *fixed* pool of LLMs. In this paper, we consider the problem of *dynamic* routing, where *new, previously unobserved* LLMs are available at test time. We propose UniRoute, a new approach to this problem that relies on representing each LLM as a *feature vector*, derived based on predictions on a set of representative prompts. Based on this, we detail two effective instantiations of UniRoute, relying on *cluster-based* routing and a *learned cluster map* respectively. We show that these are estimates of a theoretically optimal routing rule, and quantify their errors via an excess risk bound. Experiments on a range of public benchmarks show the effectiveness of UniRoute in routing amongst more than 30 unseen LLMs.

1 Introduction

Large language models (LLMs) have seen a flurry of recent development [Radford et al., 2018, 2019, Brown et al., 2020, Touvron et al., 2023, Anil et al., 2023, Grattafiori et al., 2024, DeepSeek-AI et al., 2024]. These impressive abilities notwithstanding, the inference cost of LLMs can be prohibitive [Li et al., 2024a, Wan et al., 2024, Zhou et al., 2024b]. This has motivated several techniques to improve LLM inference efficiency, such as speculative decoding [Stern et al., 2018, Chen et al., 2023a, Leviathan et al., 2023], early-exiting [Schuster et al., 2022], quantisation [Chee et al., 2023], compression [Frantar and Alistarh, 2023, Agarwal et al., 2024, Rawat et al., 2024], and others [Pope et al., 2023, S et al., 2024, Menghani, 2023].

Our interest is in *model routing* for efficient inference. Here, one maintains a pool of candidate LLMs of various sizes and capabilities. Given a prompt, one learns to predict the lowest-cost LLM which can reasonably address the prompt. In doing so, one can learn to use high-cost LLMs sparingly, only on the (relatively) few “hard” inputs. This is a conceptually simple but effective technique, which has seen a surge of recent interest [Hendy et al., 2023, Hari and Thomson, 2023, Ding et al., 2024, Sakota et al., 2024, Chen et al., 2024b, Hu et al., 2024b, Shnitzer et al., 2023, Wang et al., 2023, Stripelis et al., 2024, Ong et al., 2025, Zhuang et al., 2024, Srivatsa et al., 2024, Feng et al., 2024, Lu et al., 2024, Zhao et al., 2024b, Dann et al., 2024, Aggarwal et al., 2024, Lee et al., 2024a, Mohammadshahi et al., 2024, Chuang et al., 2025, Huang et al., 2025].

Existing works largely focus on routing over a *fixed* pool of LLMs. In practice, however, the pool of candidate LLMs can constantly change; e.g., older LLMs may be deprecated in favor of new, performant LLMs. To leverage such new LLMs, perhaps the simplest approach is to re-train the router. However, with frequent

Preprint. Under review.

Google提案のルーティング手法

推薦速度◎，柔軟性が高い，かつシンプル

- 中核技術がk-means法

これをほぼ流用

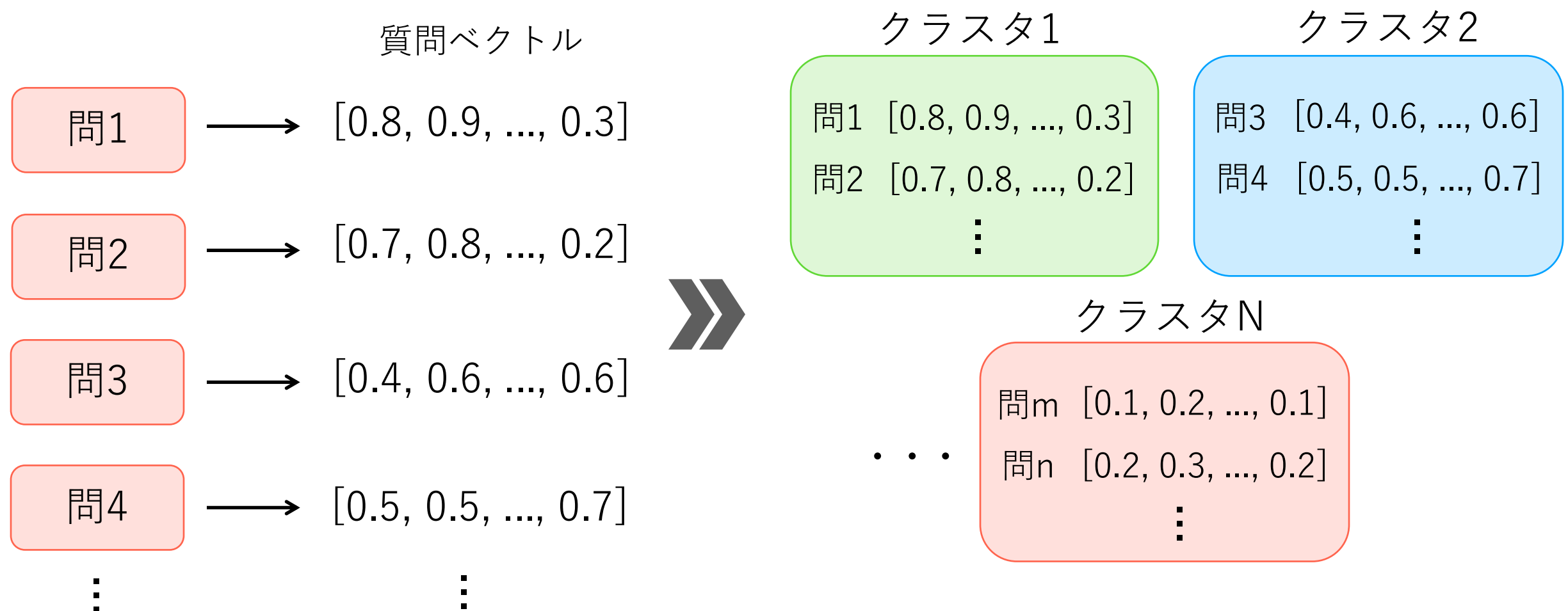
- 新規性は必ずしも求められない...？

arXiv:2502.08773v2 [cs.CL] 22 Jul 2025

アプローチ | UniRoute概要

用意した多様な質問やタスクをクラスタリング，クラスタごとにモデルを評価

- クラスタ：似た分野の質問・タスク同士の集合
- 運用時は，入力クエリの所属クラスタを算出，クラスタ内で優秀なLLMを推薦，という流れ



再現実験デモ① | ルーティング性能

75モデルのルーティング vs 単一の最優モデル

- ルーティングの効果を測るため、単一のモデルを使い続けた場合の回答性能と比較
- 比較対象： Meta-Llama-3-70B-Instruct
 - 今回の候補モデル内で、最も平均正答率が高かったモデル

結果

- UniRoute : 61.9%, Llama70B : 59.1%
 - 2.8point向上 (約4.7%の改善)
 - 推論コスト効率も改善
- ルーティングで、性能とコスト効率の向上を確認
- かなり単純化したデモでこの性能を達成

再現実験デモ② | モデル追加の影響

75モデルのルーティング vs 112モデルのルーティング

- 候補モデルを追加すると、回答性能の向上が見られるか？
- UniRoute(モデル追加前) vs UniRoute(モデル追加後)

結果

- 75モデル : 61.9%, 112モデル : 64.7%
- 2.8point向上 (約4.6%の改善)

➤ モデル追加するほど、性能向上

- モデル毎に3,000回の推論でモデル追加可能, 有効
- 最新モデルへのキャッチアップ可能性

再現実験デモ③ | 制約条件下での性能

全モデル vs 40B以下モデルのみでの精度比較

- 大きなモデル(主に70Bクラス)を常に必ずしも動かせるとは限らない
- パラメータ制約をつけた際の性能はどうなるか？

結果

- 全部 : 64.7%, 40B以下 : 61.2%, 単一最優(70B) : 59.1%
 - 3.5pointの低下 (約5.1%の悪化)
 - 一方, 単一最優モデル(llamaの70B)よりは2%↑を維持
- パラメータ制約下でもルーティングの威力を発揮
- 40B以下のモデルだけで, 70Bモデル単一を凌駕
 - 今回の実験では思いの外モデルの得意・不得意の補完が活きた

再現実験デモ③ | 制約条件下での性能

全モデル vs 40B以下モデルのみでの精度比較

- 大きなモデル(主に70Bクラス)を常に必ずしも動かせるとは限らない

ハルシネーションの低減に成功したと主張可能

- 少なくとも「誤った出力」の低減には一定の効果が見られる

ただし、複数の懸念点・改善点も見えた

- 最悪やんなくてもいいけど、やったほうがいいはいい
- オリジナリティを出す意味でもやります...

- 40B以下のモデルだけで、70Bモデル単一を凌駕
- 今回の実験では思いの外モデルの得意・不得意の補完が活きた

改善点 | 重要度高（必須級）

日本語/多言語対応

- デモでは、使用質問（ベンチマーク）は英語のみ
 - 日本語質問の際のクラスタ割り当てがかなり怪しい？
- また、embeddingモデルも英語がメイン
- 多言語対応embeddingモデルの採用，日本語データセットの検討

採用タスクの実務との乖離

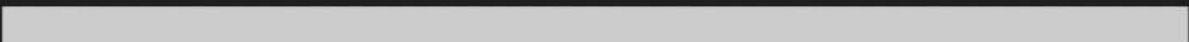
- 今回クラスタ形成に利用した質問タスクは、大半が知識を問う4択クイズ形式
- しかし、実務ではその他生成系のタスクも求められる
 - 例：コーディング，要約，翻訳，抽出型質問応答（RAG的な）
- 実用的面を意識したデータセットの検討

改善点 | 重要度高（必須級）

学習データ（ベンチマーク質問）の前処理

- 一部クラスタが、質問の表面的な形式に依存しすぎている？
- 例：クラスタNo.12 内のベンチマーク分布（truthfulQAだけで構成）

```
Detailed Category Distribution (Unique Questions Only):
```

```
-----  
1. truthfulqa_mc1          743 (100.0%) 
```

```
-----  
Total categories: 1
```

- 原因：Few-shotプロンプトになっている（例示が毎回固定）

もう少し意味的にクラスタリングできた方が良い...？

- 形式依存のクラスタになりすぎないように前処理してからk-meansなど

実験計画のイメージ

Phase 1：商用モデル vs OSSモデル 誤答率の計測

- 「ハルシネーション等による誤った出力」の検証
- OSSモデルは全般的に、商用モデルよりも誤答率が高いことを定量的に提示
- 目的：生成AI利活用リスクの特定・提示

Phase 2：ベンチマークでのルーティング実験

- 「ルーティングにより、誤回答は減少するか？」の検証
- 実験設定を整えた上で、今回のデモ実験に近いことを行う

Phase 3：実務タスクでのルーティング実験

- 「実在の社内の実務タスクでもルーティングは有用か？」の検証
- 紀陽銀行にデータ提供をしてもらう（Q&A with RAG？）

近況①

先週の滞在

- 3泊4日でQualityAI@白浜に滞在
- 会社のGPU利用とその環境(リモート)の整備が主目的
- その他, 使用ベンチマークや使用LLMの相談など

進捗

- リモートでGPU (5×5060ti) が利用できるように
- 使用ベンチマークや使用LLMが概ね決定
- 一部モデル・ベンチマークで推論テストを実行
- あとはこれを拡張していけば, 準備は整う, というところまで

近況②

Quality AI, 撤退（！？）

- 「予算，人員リソースの関係上，GENIACに参加する余裕はない，撤退せよ」との会長命令が下ったらしい
- 紀陽銀行と和歌山大学の二者間でやることに（多分）

つまり，Quality AIのGPUは使えないことに...

- 先週の滞在何やったん？
- とはいえ，基本的にコード等や再利用可能なので無駄ではなし
- 次：和歌山大学にGPUがあるみたいなのでそれで試してみる

まとめ

紀陽銀行と和歌山大学の二者間で、GENIAC-PRIZEへの応募を目指している

テーマは「生成AI利用におけるリスクの特定・低減」

AIモデルルーティングを流用し、誤回答の低減を定量的に提示

現地滞在したにも関わらず、Quality AIは撤退