LLM におけるハルシネーションの原因別検出手法の比較

中盛楓也　Yin Jou Huang　Fei Cheng
京都大学
{nakamori, huang, feicheng}@nlp.ist.i.kyoto-u.ac.jp

概要

大規模言語モデル（LLM）は高い性能を示す一方で，ハルシネーションが実社会応用の課題となっている. 本研究では, マルチホップ推論を要する HotpotQA を用い, ハルシネーションを知識不足に起因する Knowledge Gap と, 推論過程の誤りに起因する Reasoning Error に分類したデータセットを構築した. このデータセットに基づき, 代表的な 5 つのハルシネーション検出手法を比較評価した. その結果, 自然言語, ロジット, および隠れ状態に基づく手法は原因依存で性能が変動する一方，一貫性およびアテンションヘッドに基づく手法は安定した性能を示した.

1　はじめに

近年, 大規模言語モデル (LLM）は驚異的な発展を遂げ, 様々なタスクにおいて人間と同等以上の性能を示している. しかし,LLM の実社会への応用, 特に高い信頼性が求められる医療, 法律, 金融といった領域への導入において, ハルシネーションの存在が深刻な障壁となっている. ハルシネーションとは, モデルが事実と異なる内容を生成する現象である. この課題に対する有効な対抗策として, モデル自身の回答に対するハルシネーションの検出が挙げられる. 回答の正確さを事前に予測し, 信頼度が低い場合には回答を棄却したり,RAG で外部知識を検索して再生成させたりすることができる.

ハルシネーションの原因は一般に, データのノイズや曖昧性に起因する Aleatoric Uncertainty(AU) と, モデルの知識不足等に起因する Epistemic Uncertainty(EU) の 2 つに大別される [1]. 一般的な QA タスクでは質問文が明確であるため,AU の影響は小さく, 誤答の主因は EU にある. そこで本研究では, この EU をさらに, モデルに必要な知識自体が欠落していることに起因する **Knowledge Gap** と, 必要な知識は保持しているが論理推論の過程を誤ったことに起因する **Reasoning Error** の 2 つに分解する. これらの差異を明確に検証するために, 本研究ではマルチホップ QA タスクを採用する. マルチホップ QA は, 一つの質問に答えるために複数の推論ステップを経る必要がある. これにより, 各ステップの知識有無と最終的な回答の正誤の関係を分析することで, エラーが知識不足によるものか推論ミスによるものかを切り分けることが可能となる.

現在, ハルシネーション検出に関しては多様なアプローチが研究されている. これらは大きく分けて, 言語化された確信度や回答を複数回サンプリングした際の一貫性を用いるブラックボックス的な手法と, ロジット, 隠れ状態, アテンションヘッドなどのモデル内部の状態を用いるホワイトボックス的な手法に分類される [2]. しかし，既存研究の多くは，QA タスク全体に対するハルシネーション検出性能を一律に評価するにとどまっている. そのため，ハルシネーションの発生原因によって，モデルの出力に現れる挙動や指標の振る舞いがどのように異なるかについては，十分に検討されていない. 例えば，知識不足による誤答と推論過程の誤りとでは，誤りの性質が異なると考えられ，これらを区別しない評価は，検出手法の特性理解や適切な手法選択を妨げる恐れがある.

本研究では, 分類したハルシネーション原因（Knowledge Gap / Reasoning Error）ごとに, 代表的な 5 つの検出手法（自然言語, 一貫性, ロジット, 隠れ状態, アテンションヘッド）の性能評価を行い, それぞれの検出特性を比較分析する.

本研究の主な貢献は以下の通りである. 第一に,HotpotQA をサブクエスチョン単位で分解・評価し, ハルシネーションの原因（Knowledge Gap / Reasoning Error）を自動判定・付与したデータセットを作成したことである. 第二に, ハルシネーションの原因によって有効な検出手法が異なることを実験的に示ししたことである.

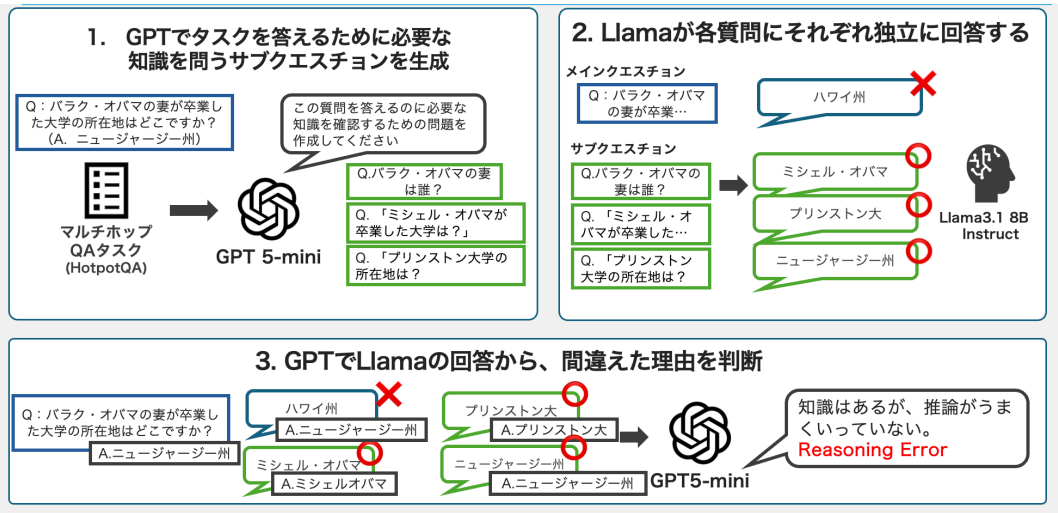

図 1　ハルシネーション原因のラベリング手法

2　データセットの作成

本節では，ハルシネーションの発生原因を Knowledge Gap，Reasoning Error，および No Mistake に分類するためのデータセット構築手順を説明する. 全体の流れを図 1 に示す.

本研究では，複数の推論ステップを要するマルチホップ QA データセットとして HotpotQA [3] を採用した.HotpotQA には，複数の事実を比較して判断する Comparison 型の質問と，複数の知識を段階的につなぐ Bridge 型の質問が含まれている. また,HotpotQA には推論過程において必要となる根拠知識を含む Supporting Facts が含まれている. しかし，Supporting Facts は解答に必要な情報を含む Wikipedia 上の文を集めたものであり，各情報間の関係や利用順序は明示的に定義されていない. そこで本研究では，モデルが回答に必要な各知識を個別に保持しているかを検証するため，Supporting Facts を入力としてサブクエスチョンを生成した. 各サブクエスチョンは，単一の知識のみを問う最小単位の質問として設計し，GPT-5 mini を用いて自動生成した. なお，HotpotQA に含まれるマルチホップを要さない質問は，本分析の対象外として除外した. 生成されたサブクエスチョンの妥当性を検証するため，先頭 100 件について人手による評価を行い，質問が適切に生成されていることを確認した.

次に，LLaMA 3.1 8B-Instruct を用いて各サブクエスチョンに対し独立に 5 回ずつ回答を生成し，3 回以上正解した場合を当該サブクエスチョンに正答したと判定した. 最終的な分類は以下のルールに基づいて行った. まず，LLaMA 3.1 8B-Instruct が元の HotpotQA の質問（メインタスク）に正解した事例を No Mistake とした. 一方，メインタスクに不正解であった事例について，構成するすべてのサブクエスチョンに正答していた場合は，必要な知識を保持しているにもかかわらず推論に失敗したとみなし Reasoning Error に分類した. これに対し，一つでも不正解のサブクエスチョンが存在した場合は，前提知識の欠如に起因すると判断し Knowledge Gap に分類した.

この手順により構築された,2000 件中マルチホップを要さない問題を除いた 1638 件の内訳を図 2 に示す. この結果から, マルチホップ QA においては, 推論ミスよりも前提知識の欠如に起因する誤答である Knowledge Gap が大半を占めていることがわかる.

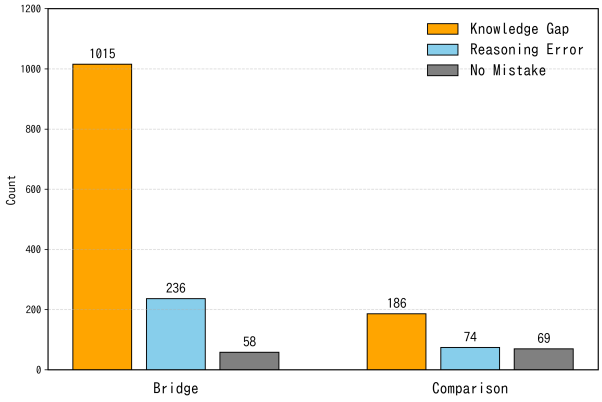

図 2　ハルシネーション原因の内訳

3　ハルシネーション検出手法

本節では, 構築したデータセットを用いて, ハルシネーション原因（Knowledge Gap / Reasoning Error）

ごとにハルシネーション検出手法の性能を評価する. 対象モデルの設定, 各検出手法の概要, 評価指標および評価手順を示した上で, 原因別の検知性能の比較結果を報告する.

3.1 モデル設定

本実験では, 検証対象の言語モデルとして Llama 3.1 8B-Instruct を使用した. 推論時の生成パラメータについては,Temperature を 0.5,Top-k を 50,Max New Tokens を 64 に設定した.

3.2 ハルシネーション検出手法

ハルシネーション検出の実装には, 言語モデルのハルシネーション検出手法分析ツールである LM-Polygraph[2] を使用した. 本実験では, 以下の 5 つの手法を採用した.

- **自然言語:** モデルに自身の回答の確信度を言語化させる手法として,PTrue[4] を使用した. 回答を出力した後にその正誤を問うプロンプトを入力し,「True」と生成される確率を利用する.
- **一貫性:** 複数回答の一貫性を測る指標として, 意味的な分散を考慮した Semantic Entropy[5] を使用した. 同一の問題に対して複数の回答をサンプリングし, それらを意味的に等価なグループに分類した上でエントロピーを算出することで, 不確実性を評価する.
- **ロジット:** 出力トークンの確率分布に基づく手法として, シーケンス全体の確信度を測る Maximum Sequence Probability (MSP) を使用した. 各ステップで生成されたトークンの対数確率を合計してシーケンス全体の尤度を算出し, モデルが生成過程全体に対してどの程度自信を持っていたかを計算する.
- **隠れ状態:** モデルの内部表現（隠れ状態）を用いる手法として, 埋め込み空間におけるマハラノビス距離を用いる Mahalanobis Distance[6] を使用した. 参照データセットから得られる隠れ状態の分布を多変量正規分布として近似し, 推論時のデータがその分布の中心からどれだけ離れているかを計算する.
- **アテンションヘッド:** アテンションヘッドの重みを用いる手法として, RAUQ[7] を使用した. ハルシネーションが発生する際に特定のヘッドにおいて直前のトークンへのアテンションが低下するという現象を利用し, 信頼度を推定する.

3.3 評価手順

本実験の目的は, 前節で構築したデータセットに基づき,Knowledge Gap および Reasoning Error の 2 つのハルシネーションの発生原因ごとに各検出手法の性能評価を行うことである. 評価指標には, ハルシネーション検出の有効性を測る指標として Prediction Rejection Ratio (PRR) [8] を採用した. 本指標は，全データセットにおける正答率と，ハルシネーションである可能性が高いと推定された回答の上位 50%を除外したデータセットにおける正答率との差分として定義される. この差分をランダム棄却を基準として正規化したものである. 指標値は -1 から 1 の範囲を取り，0 はランダム棄却と同等の性能を示す. 負の値は，ランダム棄却よりも検出性能が劣ることを意味する.

- Knowledge Gap 検出用: 「Knowledge Gap（知識不足による誤答）」と「No Mistake（正答）」を混合したデータセット.Reasoning Error と実験条件を統一するため,Kanowledge Gap の件数を 310 件に制限した.
- Reasoning Error 検出用: 「Reasoning Error（推論ミスによる誤答）」と「No Mistake（正答）」を混合したデータセット.

これにより, 各手法が「知識の欠如」と「推論の誤り」に起因するハルシネーションのそれぞれに対して, どの程度高い検知性能を示すかを評価する.

3.4 結果

各ハルシネーション検出手法に対して,Reasoning Error および Knowledge Gap に基づく PRR を算出した結果を図 3 に示す. 全体として, すべての手法において Reasoning Error に対する検知性能が Knowledge Gap よりも高い傾向が確認された.

まず, 自然言語に基づく手法（PTrue）は,Reasoning Error と Knowledge Gap のいずれにおいても PRR が負の値を示しており, ランダムに近い, あるいはそれ以下の性能であることがわかる. 特に Knowledge Gap において顕著な性能低下が見られた. この結果は, 自然言語ベースの信頼度推定が, マルチホップ推論を要する問題に対して難易度が高く, ハルシネーションを十分に検知できていないことを示唆している.

同様に, 隠れ状態（Mahalanobis Distance）を用いた手法においても,Knowledge Gap に対する検知性能は

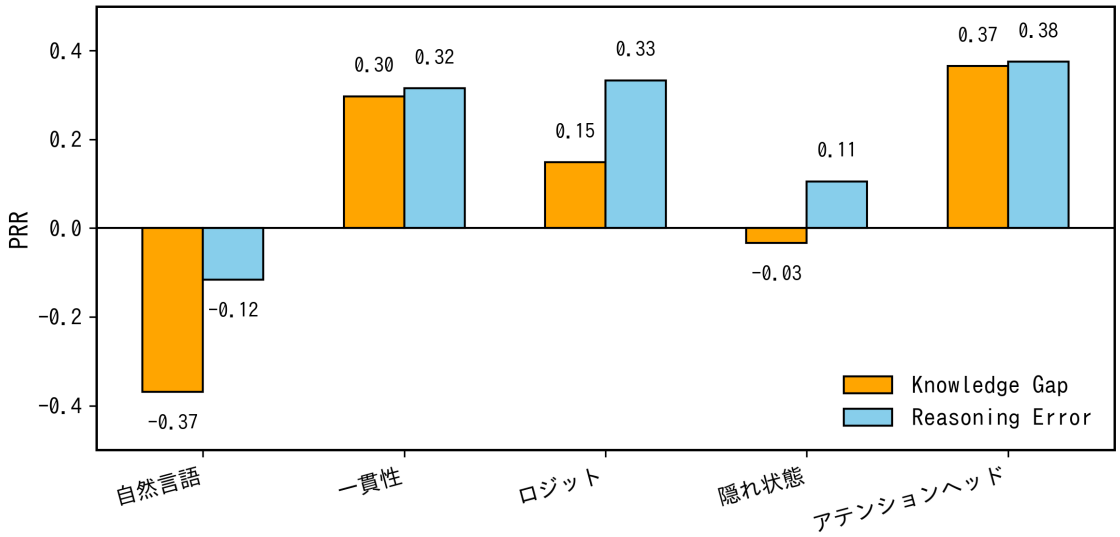

図 3　ハルシネーション原因別各手法の評価結果

低く, マルチホップ問題に対する頑健性の不足が確認された. これらの手法は, モデル内部表現を利用しているものの, 知識欠如に起因するハルシネーションを十分に捉えられていない可能性がある.

また, 自然言語に基づく手法（PTrue）, ロジットに基づく手法（Maximum Sequence Probability）, および隠れ状態に基づく手法は, Knowledge Gap と Reasoning Error の間で検知性能に大きな差が見られた. これらの手法は主に最終層の出力表現に基づいて判定を行うため, 推論過程の破綻には敏感である一方, モデル内部における知識表現の不足が顕在化しにくい Knowledge Gap に対しては十分に検知できていない可能性がある. これに対し, 一貫性に基づく手法（Semantic Entropy）およびアテンションヘッドに基づく手法（RAUQ）は, 両ハルシネーション原因においてほぼ同等の性能を示した. 特に RAUQ は, 中間層を含むアテンションヘッドの挙動を考慮することで, 知識の欠如と推論過程の誤りの双方に起因する異常を捉えていると考えられる. その結果, Reasoning Error と Knowledge Gap のいずれに対しても高い PRR を維持し, 原因に依存しない安定した検知性能を示した.

以上の結果から, ハルシネーション検出手法には, 原因（Knowledge Gap / Reasoning Error）によって検知性能が大きく変化するものと, 原因に依存せず比較的安定した性能を示すものが存在することが明らかになった. 先行研究の多くは QA タスクを一様な設定として扱い, ハルシネーション原因を区別せずに評価を行っている. しかし, 本研究の結果は, 検出手法の性能評価においてハルシネーションの原因を明示的に考慮する必要性を示しており, 今後の評価設計や手法比較において重要な観点であると言える.

4　結論

本研究では, マルチホップ QA タスクを用い, ハルシネーション原因を Knowledge Gap と Reasoning Error に分類した上で, ハルシネーション検出手法の性能比較を行った. その結果, ロジットおよび隠れ状態に基づく手法は, ハルシネーション原因によって検知性能が大きく変化する一方, 一貫性やアテンションヘッドに基づく手法は, 原因に依存せず比較的安定した性能を示すことが明らかになった. これらの結果から, ハルシネーション検出手法には, 原因によって性能が変動するものと, 変動しにくいものが存在することが示された. 先行研究では,QA タスクを一様に扱い, ハルシネーション原因を区別せずに評価を行うことが多かったが, 本研究の結果は, 検出手法の性能評価においてハルシネーション原因を考慮する重要性を示している.

今後は, 他モデルや異なる検出手法への適用を通じて, これらの傾向の一般性を検証する必要がある. ハルシネーション原因ごとの特性理解を深め, 原因に応じた検出手法の選択や統合に向けた指針を確立することが期待される.

参考文献

[1] Yasin Abbasi Yadkori, Ilja Kuzborskij, András György, and Csaba Szepesvári. To believe or not to believe your llm, 2024.

[2] Artem Shelmanov, Maxim Panov, Ekaterina Fadeeva, Artem Vazhentsev, Roman Vashurin, and Timothy Baldwin. Uncertainty quantification for large language models. In **Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics**, 2025.

[3] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering, 2018.

[4] Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know, 2022.

[5] Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In **The Eleventh International Conference on Learning Representations**, 2023.

[6] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, **Advances in Neural Information Processing Systems**. Curran Associates, Inc.

[7] Artem Vazhentsev, Lyudmila Rvanova, Gleb Kuzmin, Ekaterina Fadeeva, Ivan Lazichny, Alexander Panchenko, Maxim Panov, Timothy Baldwin, Mrinmaya Sachan, Preslav Nakov, and Artem Shelmanov. Uncertainty-aware attention heads: Efficient unsupervised uncertainty quantification for llms, 2025.

[8] Andrey Malinin, Anton Ragni, Kate Knill, and Mark Gales. Incorporating uncertainty into deep learning for spoken language assessment. In Regina Barzilay and Min-Yen Kan, editors, **Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)**, pp. 45–50, Vancouver, Canada, July 2017. Association for Computational Linguistics.

A　プロンプト全文

本節では，実験において GPT-5 mini に入力したプロンプトの全文を示す．

サブクエスチョン生成

```
You are a reasoning decomposition assistant.
Your goal is to generate the minimal set of factual sub-questions needed to answer the main multi-hop question. These sub-questions must capture all the factual information required to derive the final answer.
Guidelines: - Use the given context_info only as a reference to understand what kinds of knowledge are relevant for answering the question. (You do NOT need to check whether the knowledge is present or absent; simply use it to guide what knowledge should be asked.) - Each sub-question must test exactly one factual relation or piece of knowledge (e.g., “Who is X?”, “Where is Y located?”). - Avoid redundant or overlapping questions. - Each sub-question must explicitly include the entity names or concepts it refers to so that it can be answered independently. - Write clear, factual sub-questions that form a logical reasoning chain leading to the gold answer. - Do NOT include a final question that merely repeats the original query or asks for comparison or judgment (e.g., “Which is larger?”, “Are they the same?”). Your work ends when all factual information needed to infer the final answer has been obtained.
Output format MUST be: 1. [sub-question] → [answer] 2. [sub-question] → [answer] 3. ...
Example: Question: Where is the university located that Barack Obama’s wife graduated from? → 1. Who is Barack Obama’s wife? → Michelle Obama 2. Which university did Michelle Obama graduate from? → Princeton University 3. Where is Princeton University located? → New Jersey
Now decompose the following.
```

サブクエスチョン解答判定

```
Question: {question}
Gold Answer: {gold_answer}
Here are {num} answers generated by a model: {model_answers_json}
Task: Evaluate each model answer against the Gold Answer. Count how many of the answers are correct (semantically equivalent to the Gold Answer). Output ONLY the integer count (0 to {num}).
```