

LLM-jp FactCheck : 学習コーパスに照らした真偽検証によるモデル応答の分析

清丸 寛一¹ 出口 祥之^{2,†} 政野 美和^{1,3} 源 怜維^{1,4} 堀尾 海斗⁴櫻 惇志³ 中山 功太¹ 橘 秀幸¹ 櫻 リベカ⁵ 河原 大輔^{1,4}¹ 国立情報学研究所 大規模言語モデル研究開発センター² 奈良先端科学技術大学院大学 ³ 一橋大学 ソーシャル・データサイエンス学部⁴ 早稲田大学理工学術院 ⁵ 東京工科大学コンピュータサイエンス学部

{kiyomaru,nakayama,h_tachibana}@nii.ac.jp deguchi.hiroyuki.db0@is.naist.jp
a.keyaki@r.hit-u.ac.jp 5123053k@g.hit-u.ac.jp keyakirbk@stf.teu.ac.jp
{kakakakakaito,ray}@akane.waseda.jp dkw@waseda.jp

概要

大規模言語モデル (LLM) のモデル応答の真偽をその学習コーパスに照らして検証・分析するシステム, LLM-jp FactCheck を提案する. 本システムは, モデル応答の真偽判断のための参考情報をユーザに提示するとともに, 応答を支持する学習データの有無という観点から, モデルの知識獲得過程に関する仮説検討を可能にする. LLM-jp FactCheck を用いて LLM-jp-3 172B のモデル応答を分析した結果, 事実として正確な応答が生成される場合はそれを支持する記述が学習コーパスに含まれる一方, 事実として不正確な応答が生成される場合にそれを支持する記述は必ずしも確認されず, 誤情報の生成が必ずしも誤情報の学習に起因しないことが示唆された.

1 はじめに

大規模言語モデル (Large Language Model ; LLM) は膨大な学習コーパス上での事前学習を通じて広範な知識を獲得している [1, 2]. しかし, その知識は完全・正確とは限らず, 実際的事実と整合しない内容を生成する現象, ハルシネーションが生じることも知られている [3].

こうした背景から, LLM のモデル応答の真偽検証を行うシステムが開発されている [4, 5]. 真偽検証システムは, モデル応答中に含まれる主張や事実を切り出し, それらを支持または否定する記述を特定の情報源から検索・提示することで, ユーザに対



図 1 LLM-jp FactCheck のユーザインターフェース. 左側のビューで分析対象の LLM と対話できる. Fact-check のボタンを押下すると, 対応するモデル応答の真偽を学習コーパスに照らして検証した結果が右側のビューに表示される.

して真偽判断のための参考情報を与える.

一方で, LLM が保持する正確あるいは不正確な知識が何に由来するのか, 学習コーパスの観点から明らかにする取り組みも進みつつある. 現在までに公開されてきた LLM の多くは学習コーパスの具体的な構成が非公開であるため [6, 7], モデルが持つ知識がどのような学習データに由来するのかを直接的に分析することは困難であった. しかし近年になって学習コーパスまで含めて公開している LLM が登場し [8, 9, 10, 11], モデルの知識や能力を学習コーパスの観点から分析することが可能になってきた [12, 13].

本研究では, 真偽検証システムの枠組みに基づき, モデル応答の真偽をその学習コーパスに照らして検証するシステム **LLM-jp FactCheck** を提案する

[†] 現在, NTT 株式会社

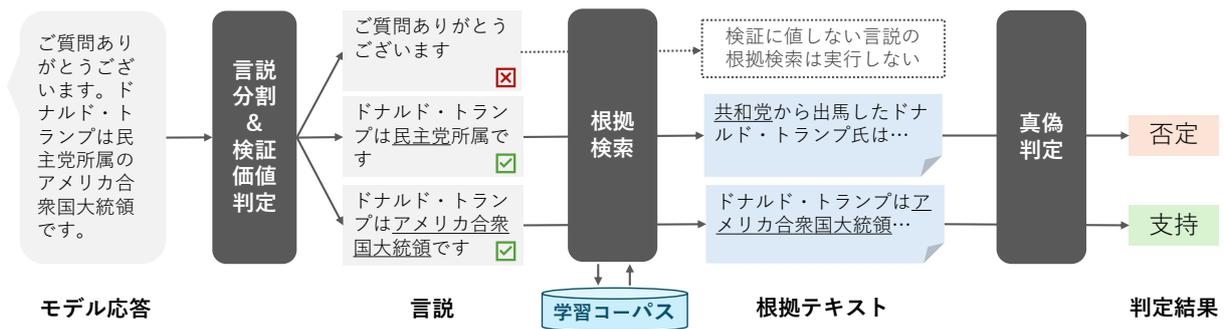


図2 LLM-jp FactCheckの概要. 分析対象のモデル応答は, 一つの事実や関係について記述した情報単位である言説に分割される. 検証に値する言説について, 学習コーパスから根拠テキストを検索・取得し, その真偽を判定する.

(図1). 本システムは, モデル応答が含む主張や事実について, それらを支持または否定する学習データ中の記述を提示することで, ユーザがモデル応答の信頼性を判断するための参考情報を提供する. 同時に, 本システムはLLMの知識獲得過程を分析するための基盤としても機能する. 例えば, 事実として正確な応答が得られるにも関わらず, それを直接的に支持する記述が学習データ中に確認できない場合には, 断片的に記述された知識の統合や推論による汎化が起きている可能性が示唆される. 反対に, 事実として不正確な応答が得られるにも関わらず, 学習コーパスには正確な記述が存在する場合には, 学習頻度や表現形式の偏りといった要因で当該知識がモデルに定着している可能性[14]や, 獲得した知識を推論時に適切に想起できなかった可能性が示唆される[15].

図2にシステムの概要を示す. システムはまずモデル応答を一つの事実や関係について記述した情報単位である言説に分割する. 次に各言説が検証に値するかを分類し, 検証に値する言説について, 学習コーパスから根拠テキストを検索・取得する. 最後に取得した根拠テキストに照らして各言説の真偽を判定する.

ケーススタディとして, 日本語質問応答データセット AI 王¹⁾に収録されている事実型質問を対象に, LLM-jp-3 172Bのモデル応答を分析した. 分析の結果, モデル応答が正解である場合には, それを支持する記述が学習コーパス中に含まれる一方, モデル応答が不正解である場合には, 必ずしもその誤回答を支持する記述は学習コーパス中に確認されず, 誤回答は必ずしも誤情報の学習に起因しないことが示唆された.

1) <https://sites.google.com/view/project-ai0/dataset>

2 関連研究

真偽検証システム 真偽検証システム[4]は, フェイクニュースに代表される, インターネット上の誤情報や偽情報を効率的に特定することを主要な目的として研究・開発が進められている. そのため, 真偽検証システムは通常, 信頼できる, 最新の知見を提供する情報源に基づき主張の真偽を判定する. 本研究の提案システムは, 情報源として分析対象のLLMの学習コーパスを用いることで, モデルが学習した範囲における真偽を検証・分析する点に特徴がある. これにより, モデルが学習コーパスからどのように知識を獲得しているかの分析が可能となる.

学習コーパス検索 学習コーパスの検索に基づきLLMの振る舞いを分析するためのシステムがいくつか提案されている. Piktusら[12]は, Pileデータセット[16]をはじめとするLLMの事前学習にしばしば使用される公開大規模コーパスの全文検索ツール, GAIA Searchを提案している. またLiuら[13]は, 部分一致文字列の高速検索アルゴリズム[17]に基づき, モデル応答に含まれる部分文字列と一致するテキスト断片を学習コーパス中から特定するシステム, OLMoTraceを提案している. 本研究で提案するシステムは, こうした学習コーパスの検索システムを拡張し, 検索結果に基づく真偽判定まで行う点に特徴がある. これにより, モデル応答に関連する学習データの有無だけでなく, 学習コーパス中の知識がモデル応答にどのように反映されているかの分析まで可能になる.

3 LLM-jp FactCheck

図2にシステム構成の概要を示す. システムはまずモデル応答を一つの事実や関係について記述し

表 1 問題「リンゴが木から落ちる様子を見て「万有引力」を発見したといわれる、17世紀のイギリスの科学者は誰？（正解：アイザック・ニュートン）」に対する LLM-jp-3 172B のモデル応答「リンゴが木から落ちる様子を見て「万有引力」を発見したといわれる、17世紀のイギリスの科学者はアイザック・ニュートンです。ニュートンは「万有引力の法則」を確立し、近代物理学の基礎を築きました。」の分析結果。

言説	根拠テキスト（抜粋）	判定
リンゴが木から落ちる様子を見て『万有引力』を発見したといわれる、17世紀のイギリスの科学者はアイザック・ニュートンです。	… 17世紀から18世紀にかけてニュートンはりんごが木からポトッと落ちるのを見て、万有引力の法則を発見したという有名な逸話（いつわ）がある。…	支持
アイザック・ニュートンは『万有引力の法則』を確立しました。	アイザック・ニュートン… 万有引力を発見した話は広く知られている…	支持
アイザック・ニュートンは近代物理学の基礎を築きました。	… アイザック・ニュートンさん… ニュートン力学を確立し、古典力学や近代物理学の祖となった。…	支持

表 2 問題「映画『ウエスト・サイド物語』に登場する2つの少年グループといえば、シャーク団と何団？（正解：ジェット団）」に対する LLM-jp-3 172B のモデル応答「映画『ウエスト・サイド物語』に登場する2つの少年グループは、「シャーク団」と「ジェット団」です。…」の分析結果。

言説（抜粋）	根拠テキスト（抜粋）	判定
映画『ウエスト・サイド物語』に登場する2つの少年グループはシャーク団とジェット団です。	… {"qid": "AIO02-0001", "competition": "第2回 AI 王", "timestamp": "2021/01/29", "section": "開発データ問題", "number": 1, "original_question": "映画『ウエスト・サイド物語』に登場する2つの少年グループといえば、シャーク団と何団?", "original_answer": "ジェット団", …	支持

た情報単位、言説に分割する。このとき、言説はそれ単体で内容が理解できるよう、代名詞を照応先の名詞で置換するなど、必要に応じて情報が補完される。言説単位で真偽検証を行うことには主に二つの目的がある。一つ目は解釈性の向上である。言説ごとに真偽検証することで、モデル応答のどの部分が正確でどの部分が不正確かを詳細に把握できる。二つ目は検証可能性の向上である。後段の根拠テキストを検索する処理において、モデル応答全体を検索クエリとして用いると、しばしば複数の事実や関係が一つのクエリに混在することになり、検索結果の関連性が低下しやすい。その結果、取得した根拠テキストに基づいてモデル応答の真偽を判定することが困難になる。言説単位で検索を行うことで、検索の焦点を単一の事実や関係に限定でき、真偽判定に有用な根拠テキストを取得しやすくなる。

次に、真偽検証を行う言説を選別するため、各言説が検証に値するかを分類する。言説が何らかの情報源に基づき真偽を客観的に判断できる事柄を表す場合、検証に値すると分類する。言説が疑問や命令、挨拶、主観的な意見など、外部の情報源に基づき真偽を検証できない内容の場合、検証に値しないと分類する。

検証に値する言説について、分析対象のモデルの学習コーパスから根拠テキストを検索し、得られた

根拠テキストに照らして真偽を判定する。言説が根拠テキストに含意される内容であれば支持、矛盾する内容であれば否定、判断がつかない内容であれば不明のラベルを与える。ここでの真偽判定は、実際の事実に照らしての真偽ではなく、あくまで取得した根拠テキストに照らしての真偽に着目していることに注意されたい。例えば「地球は平らだ」という実際の事実とは異なる言説であっても、取得した根拠テキストにその事実を支持する言及が含まれていれば、システムは支持のラベルを与える。

4 LLM-jp-3 172B の応答分析

ケーススタディとして、LLM-jp FactCheck を利用し、LLM-jp-3 172B のモデル応答を分析した。LLM-jp-3 172B はおよそ 2.1 兆トークンの学習コーパスで学習された約 1,720 億パラメータの LLM である。本モデルは、モデルパラメータ、学習コーパス²⁾ともに一般に公開されている。ケーススタディでは、質問に対する回答を安定して生成させるため、指示チューニング済みモデル³⁾を使用した。

モデルへの入力として、日本語質問応答データセット AI 王 Version 2.0 開発用データ¹⁾の問題を使

2) <https://gitlab.llm-jp.nii.ac.jp/datasets/llm-jp-corpus-v3>

3) <https://huggingface.co/llm-jp/llm-jp-3-172b-instruct3>

表 3 問題「日本のテレビ業界で「プライムタイム」といえば、毎日 19 時から何時までの時間帯のこと？（正解：23 時）」に対する LLM-jp-3 172B のモデル応答「… 一般的には 19 時頃から 22 時頃までを指すことが多いです。」の分析結果。

言説（抜粋）	根拠テキスト（抜粋）	判定
一般的にはプライムタイムは 19 時頃から 22 時頃までを指すことが多い。	… プライムタイムとは、… 一言で説明すると『1 日のうちで視聴率が最も高くなる時間帯』のことで、日本の場合、毎日 19 時～23 時の時間帯を指す。…	否定

用した。本データセットは事実型の質問を中心に構成されており、検索に基づく真偽検証と高い親和性を持つ。

4.1 実装の詳細

言説分割、検証価値判定、真偽判定はいずれも gpt-4o (gpt-4o-2024-05-13) のプロンプティングによって行った。使用したプロンプトを A 節に示す。

学習コーパスの検索は BM25 [18] に基づく疎ベクトル検索によって行った。実装には Elasticsearch⁴⁾ を使用した。検索インデックスを構築する前処理として、LLM-jp-3 172B のトークナイザーを用いて、文書を最大 300 トークンのチャンクに分割した。言説の根拠テキスト検索では、言説のテキストそのものを検索クエリとし、関連度上位 3 件のチャンクを得た。

4.2 モデル応答の分析

AI 王 Version 2.0 開発用データに収録されている問題に対する LLM-jp-3 172B のモデル応答から、分析対象として正解応答と不正解応答をそれぞれ 10 件ずつ抽出した。これらの応答について LLM-jp FactCheck で検証を行い、正解・不正解に至った要因を分析した。

正解応答を分析した結果、10 件すべてでモデルの回答を支持する文書が学習コーパス中に存在することを確認した。表 1 に正解応答の分析例を示す。分析した正解応答では全て、本例のように学習データ中に正解応答を明確に支持する言及が存在していた。この結果は、分析対象の正解応答が、断片的な知識の統合や汎化というよりも、当該知識に関する直接的な記述を学習したことに基づき生成された可能性を示唆している。

また、正解応答を分析する中で、AI 王 Version 2.0 開発用データのデータリークが存在することが確認された。表 2 にその例を示す。本例で得られた根拠テキストは AI 王への参加報告記事の一部であり、

Common Crawl⁵⁾ 由来の日本語 Web コーパスに含まれていた。このデータリークは、本例でモデルが正解応答を生成できたことに一定程度寄与した可能性がある [19]。

不正解応答の分析では、10 件すべてで不正解応答を支持する文書の存在は確認されなかった⁶⁾。また、10 件中 8 件では誤ったモデル応答を否定する、事実として正しい文書が学習コーパス中存在することが確認された。表 3 に不正解応答の分析例のうち、事実として正しい文書が学習コーパス中に存在した例を示す。このような例では、学習コーパス中に正解に関する記述が存在していることから、誤答の要因は学習データの不在ではなく、当該知識がモデルに定着していなかったこと、あるいは獲得した知識を推論時に適切に想起できなかったことにあると考えられる [15]。

5 まとめ

本研究では、モデル応答の真偽を分析対象の LLM の学習コーパスに照らして検証するシステム、LLM-jp FactCheck を提案した。LLM-jp-3 172B の事実型質問に対するモデル応答を分析し、事実として正確なモデル応答は当該事実を支持する学習データの存在と強く対応する一方で、事実として不正確なモデル応答は必ずしも誤情報の学習に起因しないことを実験的に確認した。

今後の展望として、密ベクトル検索による根拠テキストの検索が挙げられる。ケーススタディでは BM25 に基づく疎ベクトル検索を用いたが、密ベクトル検索を用いることで、表層的には異なるものの意味的に関連する根拠テキストを取得できる可能性がある。これにより、言説を支持・否定する学習データの存在をより精緻に分析できると期待される。また、その際に多言語埋め込みを用いることで、知識の言語間転移の分析も可能になると考えられる。

5) <https://commoncrawl.org/>

6) 学習コーパス中に不正解応答を支持する文書が存在しないことを保証するわけではない。

4) <https://www.elastic.co/elasticsearch>

謝辞

本研究成果の一部は、データ活用社会創成プラットフォーム mdx を利用して得られたものである。

参考文献

- [1] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In **Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing**, 2019.
- [2] Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing**, 2020.
- [3] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. **ACM Transactions on Information Systems**, Vol. 43, No. 2, pp. 1–55, 2025.
- [4] Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. A survey on automated fact-checking. **Transactions of the Association for Computational Linguistics**, Vol. 10, , 2022.
- [5] Yuxia Wang, Revanth Gangi Reddy, Zain Muhammad Mujahid, Arnab Arora, Aleksandr Rubashevskii, Jiahui Geng, Osama Mohammed Afzal, Liangming Pan, Nadav Borenstein, Aditya Pillai, Isabelle Augenstein, Iryna Gurevych, and Preslav Nakov. Factcheck-bench: Fine-grained evaluation benchmark for automatic fact-checkers. In **Findings of the Association for Computational Linguistics: EMNLP 2024**, 2024.
- [6] Xin Liu, Daniel McDuff, Geza Kovacs, Isaac Galatzer-Levy, Jacob Sunshine, Jiening Zhan, Ming-Zher Poh, Shun Liao, Paolo Di Achille, and Shwetak Patel. Large language models are few-shot health learners. **arXiv preprint arXiv:2305.15525**, 2023.
- [7] Llama Team, Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. **arXiv preprint arXiv:2407.21783**, 2024.
- [8] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O’ Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff ほか. Pythia: A suite for analyzing large language models across training and scaling. In **International Conference on Machine Learning**, pp. 2397–2430. PMLR, 2023.
- [9] Dirk Groeneveld, Iz Beltagy, Evan Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. Olmo: Accelerating the science of language models. In **Proceedings of the 62nd annual meeting of the association for computational linguistics**, pp. 15789–15809, 2024.
- [10] LLM-jp, Akiko Aizawa, Eiji Aramaki, Bowen Chen, Fei Cheng, Hiroyuki Deguchi, Rintaro Enomoto, Kazuki Fujii, Kensuke Fukumoto, Takuya Fukushima, Namgi Han, et al. Llm-jp: A cross-organizational project for the research and development of fully open japanese llms. **arXiv preprint arXiv:2407.03963**, 2024.
- [11] Zhengzhong Liu, Aurick Qiao, Willie Neiswanger, Hongyi Wang, Bowen Tan, Tianhua Tao, Junbo Li, Yuqi Wang, Suqi Sun, Omkar Pangarkar, et al. Llm360: Towards fully transparent open-source llms. **arXiv preprint arXiv:2312.06550**, 2023.
- [12] Aleksandra Piktus, Odunayo Ogundepo, Christopher Akiki, Akintunde Oladipo, Xinyu Zhang, Hailey Schoelkopf, Stella Biderman, Martin Potthast, and Jimmy Lin. GAIA search: Hugging face and pysnerini interoperability for NLP training data exploration. In **Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics**, 2023.
- [13] Jiacheng Liu, Taylor Blanton, Yanai Elazar, Sewon Min, Yen-Sung Chen, Arnavi Chheda-Kothary, Huy Tran, Byron Bischoff, Eric Marsh, Michael Schmitz, et al. Olmo-trace: Tracing language model outputs back to trillions of training tokens. In **Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics**, pp. 178–188, 2025.
- [14] Kushagra Bhushan, Yatin Nandwani, Dinesh Khandelwal, Sonam Gupta, Gaurav Pandey, Dinesh Raghu, and Sachindra Joshi. Systematic knowledge injection into large language models via diverse augmentation for domain-specific RAG. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, **Findings of the Association for Computational Linguistics: NAACL 2025**, 2025.
- [15] Lei Yu, Meng Cao, Jackie CK Cheung, and Yue Dong. Mechanistic understanding and mitigation of language model non-factual hallucinations. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, **Findings of the Association for Computational Linguistics: EMNLP 2024**. Association for Computational Linguistics, 2024.
- [16] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. **arXiv preprint arXiv:2101.00027**, 2020.
- [17] Jiacheng Liu, Sewon Min, Luke Zettlemoyer, Yejin Choi, and Hannaneh Hajishirzi. Infini-gram: Scaling unbounded n-gram language models to a trillion tokens. **arXiv preprint arXiv:2401.17377**, 2024.
- [18] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. **Okapi at TREC-3**. 1995.
- [19] Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In **Findings of the Association for Computational Linguistics: EMNLP 2023**, 2023.

A プロンプト

根拠テキストの検索を除くテキスト理解を要する処理は全て gpt-4o のプロンプティングで実装した。言説分割のシステムプロンプト⁷⁾を示す。

You are provided with a document (or an utterance), which may include optional context. Your task is to decompose the document into individual, atomic claims. Each claim should represent a single fact and must be comprehensible on its own, without the need for additional context. For instance, pronouns should be replaced with the specific names they refer to when formulating each claim.

Example:

Input:

Context: What do you know about Mary?

Document: She likes playing piano and doesn't like cookies.

Output:{"claims": ["Mary likes playing piano.", "Mary doesn't like cookies."]}

Example:

Input:

Context: アメリカの初代大統領は誰ですか？

Document: ジョージ・ワシントンです。

Output:

{"claims": ["アメリカの初代大統領はジョージ・ワシントンです。"]}

Note:

The claims should be in the same language as the document.

検証価値判定のシステムプロンプトを示す。

You are provided with claims. Your task is to determine whether each claim should be fact-checked. For example, subjective claims, trivial facts, and questions are not check-worthy.

Example:

Input:

- Friends is a great TV series.

- The Stanford Prison Experiment was conducted in the basement of Encina Hall.

- I think Apple is a good company.

- Are you sure Preslav is a professor in MBZUAI?

- As a language model, I can't provide these info.

Output:

{"labels": [true, true, false, false, false]}

真偽判定のシステムプロンプトを示す。

You are provided with a claim and a list of evidences. Your task is to determine whether the evidences support the claim. Return true if the evidences support the claim. Return false if the evidences contradict the claim. Return null if the claim cannot be verified or falsified.

Example:

Input:

Claim: The earth is flat.

Evidences:

1. The earth is round.

2. Some people believe that the earth is flat, but they are wrong.

Output:

{"rationale": "The claim is not supported by any of the evidences.", "label": false}

7) 処理対象の入力はユーザープロンプトとして与えた。この他の処理についても同様である。