意味的類似度と NLI による候補絞り込みを用いた RAG データベースのレコード間矛盾検知

中島 明 [1]　田宮寛人 [1]　柿崎和也 [1]
[1] 日本電気株式会社
{akira-nakashima,hiroto-tamiya,kazuya1210}@nec.com

概要

Retrieval-Augmented Generation（RAG）は，大規模言語モデル（Large Language Model，LLM）と外部知識を組み合わせた応答生成手法であるが，参照データベースに矛盾が含まれる場合，誤った応答が生成される恐れがある．データベースのレコード間の矛盾判定において，高い文脈理解能力を持つ矛盾検知に LLM を用いることは有望であるが，候補対数の増大に伴うトークン消費が課題となる．本稿では，類似度に基づいて選出された候補ペアに対して，自然言語推論（Natural Language Inference，NLI）モデルを応用した矛盾スコア算出によるフィルタリングを導入することで，検知性能を維持したまま LLM の入出力トークン数を削減する手法を提案する．また，擬似データベースを用いた実験により，ベースライン手法と同等以上の F1 スコアを保ちつつ，総トークン数を 70%未満に削減可能であることを示す．

1　はじめに

大規模言語モデル（Large Language Model，LLM）は，高い文脈理解能力を持ち多様なタスクを実行することが可能である．しかし，LLM の応答は内部パラメータに保持される知識だけに依存するため，学習時点以降の最新情報や組織内のドメイン固有知識を反映できない [1, 2]．この課題に対する代表的な解決策として，外部の情報源から関連文書を検索し，その内容をプロンプトに挿入して応答を生成する RAG が注目されている [3, 4]．RAG では，文書をチャンク分割し，その埋め込みベクトルを検索に利用するベクトルデータベースが一般的に採用されている [5]．

一方で，RAG で参照されるベクトルデータベースは，知識の更新や誤植，誤情報の混入により，必ずしも一貫性のある状態に保たれているとは限らない [6]．そうした矛盾を含むデータソースを用いると，LLM は矛盾した証拠文書に基づいて回答を生成することになり，不正確な回答を行う可能性が高くなる [7]．したがって，RAG システムの信頼性を確保するためには，ベクトルデータベース自体の整合性を維持し，矛盾を含むレコードを検出し修正することが重要である．

しかし，レコード数を n とすると検査すべきレコード対は $O(n^2)$ に増加する．そのため，大規模なデータベースに対して総当たりで矛盾検知を行うことは困難である．特に，矛盾判定に商用 LLM API を利用する場合，入出力トークン数に応じて課金額が決定されるため，トークン数の削減は運用コスト低減に直結する [8]．したがって，検知精度を維持しながら，LLM の入出力トークン数を削減する仕組みが必要である．

検査対象とする候補ペアの絞り込みとして，埋め込みベクトルの類似度に基づく近傍探索やクラスタリングなどが利用できる [9]．しかし，候補ペアの中に明らかな非矛盾ペアが多く含まれる場合，LLM 判定に無駄なトークンが消費されることになる．

本稿では，LLM へ検査させる候補ペアに対して，自然言語推論（Natural Language Inference，NLI）モデルを利用したフィルタリングを導入する．提案手法を図 1 に示す．提案手法では，候補ペアに対して，NLI ベースのフィルタリングを行い矛盾している可能性が高いペアを抽出し，LLM で判定を行う二段階の絞り込みの構成をとる．この構成により，矛盾検知精度を維持しながら，LLM の入出力トークン数を削減できることを示す．

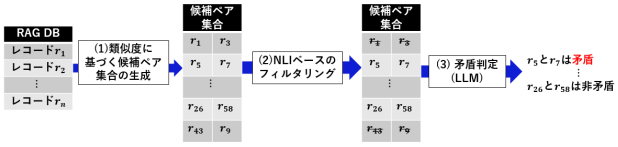

図 1　データベース全体のレコード間矛盾検知に対する提案手法の概要．Top-k 近傍探索またはクラスタリングにより候補生成後，候補ペアを文分割して NLI 矛盾スコアを算出し，閾値を超える候補のみを LLM で最終判定する．

2　関連研究

2.1　RAG における矛盾検知

LLM が直面する矛盾は，内部知識同士の矛盾（intra-memory conflict），外部知識と内部知識の矛盾（context-memory conflict），外部知識同士の矛盾（inter-context conflict）に 3 種類に分類される [10]．RAG におけるデータソース内の矛盾は inter-context conflict の一例として位置づけられる．

RAG の回答生成時に，LLM へ矛盾する証拠が与えられた際に，誤った証拠を根拠なく採用したり内部知識のみに基づいて回答するなど，不安定な振る舞いが引き起こされることが報告されている [7]．また，RAG や検索を利用した LLM システムにおいて，矛盾タイプ毎とタイプ毎の望ましい対処を体系化し，矛盾するソースが与えられた際のタイプ毎の挙動を分析した研究も行われている [11]．さらに，取得されたコンテキストが非矛盾かの検証器としての LLM 利用も提案されている [12]．矛盾した取得文書への対策手法として，各文書の確からしさを複数エージェントに議論させ，統合した回答を生成する手法も提案されている [13]．

このように RAG の参照先データに矛盾が含まれうることは問題として広く認識されているが，既存研究は主に，矛盾したコンテキストを与えた際の LLM の挙動の分析や，回答生成時の矛盾対処に焦点が当てられている．本稿では，クエリを介さずに，ベクトルデータベース全体を対象としてレコード間矛盾を洗い出す設定を扱う．

2.2　NLI

NLI は，前提文と仮説文の関係を含意・矛盾・中立に分類するタスクである [14, 15]．近年では，BERT[16] などの事前学習モデルをベースにした，数百 M 程度のパラメータ数のモデルが高い分類性能を挙げている [17, 18]．しかし，既存の NLI モデルは，複数文からなる長い入力にそのまま適用すると性能が低下し得る．そこで，複数文からなる長い入力に対して，文ペア単位の NLI スコアを集約して整合性を判定する手法が提案されている [19, 20, 21]．本稿においても，複数文からなるテキストに NLI を適用する際に文単位でのスコア算出を利用する．

3　提案手法

提案手法は，(1) 意味的類似度に基づく候補ペア生成，(2)NLI ベースの矛盾スコア計算に基づくペアのフィルタリング，(3)LLM による矛盾判定，の 3 つの処理からなる．

3.1　タスクの設定

本稿では，複数の文書レコード r_i が並んだレコード集合 $D = \{r_1, \ldots, r_n\}$ を対象として，互いに矛盾するレコード対の集合 $C = \{(r_i, r_j) \mid r_i \text{ と } r_j \text{は矛盾}\}$ の抽出を目標とする．

3.2　類似度に基づく候補ペア生成

総当たりで全ペアを比較する場合，比較対象のペアは $\binom{n}{2} = n(n-1)/2$ 個となる．そこで，意味的に類似したレコード同士をペア候補として抽出し，候補ペア集合を構成する．意味的に類似したレコード対の抽出方法として，埋め込みベクトルを利用した近傍探索および K-means 等のクラスタリング手法を利用できる．

Top-k　各レコード r_i に対して，埋め込み類似度に基づく近傍上位 k 件を取得する．それらを r_i の比較対象として，重複が含まれないように候補に追加する．このとき，候補数はおよそ $n \cdot k$ となる．

クラスタリング　レコード集合を埋め込み空間上で K-means 等によりクラスタリングし，同一クラスタ内の組み合わせペアを候補とする．候補数は，クラスタ数 K に対しておよそ n^2/K となる．

3.3　NLI による候補フィルタリング

候補ペアに対して，NLI モデルを利用してレコード対 (r_i, r_j) の矛盾スコアを算出し，閾値判定を行う．ただし，既存の NLI モデルは短文ペアを中心に学習されているため，レコード r_i, r_j をそれぞれ文単位に分解した $\{s_{i_1}, \ldots, s_{i_p}\}$ および $\{s_{j_1}, \ldots, s_{j_q}\}$ について，各文ペア (s_{i_a}, s_{j_b}) に NLI を適用し，矛盾確率を計算する．それらの最大値を取り，それをレ

コード対 (r_i, r_j) の矛盾スコアとし，閾値 θ を超えたペアのみを LLM の矛盾判定対象とする．

3.4 LLM による矛盾判定

前段フィルタを通過した候補ペア (r_i, r_j) に対して，LLM を用いて矛盾判定を行う．ここでは，レコードのペアをプロンプトとして LLM に提示し，矛盾しているか否かの二値判定を出力するよう指示を与える．矛盾と判定されたレコードペアを検出結果として出力する．矛盾判定用のプロンプトを図 2 に示す．

Prompt

次の 2 つの文章に関して矛盾はありますか？
1. {text1}
2. {text2}
矛盾している場合は 'YES'、そうでない場合は 'NO' と答えてください。

図 2 矛盾判定用のプロンプト．

4 実験

4.1 データセット

実験には，日本語 RAG 向け評価用データセットである JOaRA（Japanese Question Answering with Retrieval Augmentation）[1] を基に作成したレコード集合を用いた．JOaRA は，質問と回答および参照文書からなるレコードで構成される．我々は，参照文書から 250 件をランダムにサンプリングし，各レコードから元の内容と矛盾する改変レコードを `gpt-4o`[2] を用いて作成することで，500 件のレコードからなる矛盾を含む疑似データベースを構築した．オリジナルのレコードと対応する改変レコードを正解の矛盾ペアとした．

4.2 使用モデル

実験では，LLM は `gpt-4o-mini`[3] を温度パラメータを 0 に指定して利用し，文埋め込みは多言語埋め込みモデル `multilingual-e5-large`[4] を用いた．

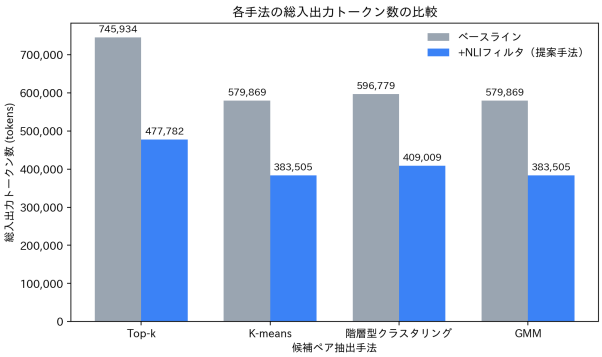

図 3 500 件のレコードを対象にした矛盾検知に関する候補生成法ごとの LLM 総入出力トークン数．ベースライン（候補生成 →LLM）は灰色のグラフ．提案手法（候補生成 →NLI フィルタ →LLM）は青色のグラフ．

NLI モデルは，日本語 NLI データセット JSNLI[5]．で事前学習された `bert-base-japanese-jsnli`[6] を用いた．

4.3 候補生成

ベースライン 埋め込みベクトルの類似度に基づいて候補ペアを生成し，各ペアに対して LLM を用いて矛盾判定を行う手法をベースラインとする．候補ペアの生成方法として，各レコードの Top-k 近傍を候補とする方法（$k = 5$）と，埋め込みベクトルをクラスタリングし，同一クラスタ内のレコード対を候補とする方法を用いた．クラスタリング手法として，K-means，階層型クラスタリング，混合ガウスモデル（GMM）を利用した．クラスタリングの実装には scikit-learn ライブラリを用い，各クラスタにおよそ k 件のレコードが含まれるようにクラスタ数を設定した．

提案手法 ベースラインと同様にして，埋め込みベクトルに基づいて選出された候補ペアについて，NLI モデルを利用した矛盾スコアを算出し，閾値 θ 以上のペアのみを LLM に入力し矛盾判定を行う．本実験においては，$\theta = 0.7$ を利用した．

4.4 評価指標

評価指標として，レコード全体の矛盾検知処理における，LLM の総入出力トークン数および矛盾判定に関する precision, recall, F1 値を利用する．入出力トークン数については，OpenAI の提供する tiketoken

1) https://github.com/hotchpotch/JQaRA
2) https://platform.openai.com/docs/models/gpt-4o
3) https://platform.openai.com/docs/models/gpt-4o-mini
4) https://huggingface.co/intfloat/multilingual-e5-large
5) https://nlp.ist.i.kyoto-u.ac.jp/index.php?日本語SNLI(JSNLI)データセット
6) https://huggingface.co/Formzu/bert-base-japanese-jsnli

表 1　矛盾検知の性能比較．各候補生成法に対するベースライン（候補生成 →LLM）と提案手法（候補生成 →NLI フィルタ →LLM）の precision，recall，F1．

手法	Precision	Recall	F1
Top-k → LLM	0.35	0.95	0.51
Top-k → NLI フィルタ → LLM(提案手法)	**0.43**	0.90	**0.58**
K-means → LLM	0.35	0.87	0.50
K-means → NLI フィルタ → LLM (提案手法)	**0.43**	0.82	**0.57**
階層型クラスタリング → LLM	0.35	0.90	0.52
階層型クラスタリング → NLI フィルタ → LLM (提案手法)	**0.44**	0.90	**0.59**
GMM → LLM	0.35	0.87	0.50
GMM → NLI フィルタ → LLM (提案手法)	**0.43**	0.82	**0.57**

ライブラリ[7)]を使用し，利用モデルに対応するトークナイザでの入力プロンプトと出力プロンプトの合計トークン数を計算している．

4.5　結果

総入出トークン数　図 3 に，各候補生成手法における総使用トークン数を示す．提案手法は，ベースラインと比較して，総トークン数が 64.1% から 68.5% に削減されている．いずれの候補ペア生成方法においても，LLM が利用する総トークン数がおよそ同様の割合に削減されていることがわかる．

矛盾検出精度　表 1 に，ベースラインと提案手法の precision，recall，F1 スコアを示す．入出力トークン数の削減にもかかわらず，提案手法とベースラインでは，矛盾判定性能はほとんど変化していないことがわかる．Recall の低下は最大でも 0.05 である一方で，全ての候補生成法で precision が向上し，結果として F1 スコアはベースライン手法より向上している．これは，NLI を用いたフィルタリングによる矛盾ペアの取りこぼしはわずかである一方で，非矛盾ペア除去によって LLM の誤検知の抑制が生じた結果だと思われる．

5　おわりに

本稿では，RAG の参照先データベースを対象として，レコード間の矛盾を検出する手法について検討した．具体的には，埋め込みに基づく近傍探索またはクラスタリングで候補ペアを生成した後，NLI を利用したフィルタリングを挿入し，矛盾可能性が高い候補のみを LLM で判定する構成の矛盾検知パイプラインを提案した．この構成により，商用 LLM API の利用時にコスト面で重要となる総入出力トークン数を削減しつつ，矛盾検知性能を維持できることを示した．

実験では，JOaRA を基に構築した矛盾を含んだ擬似的なデータベースを用い，Top-k 近傍探索および複数のクラスタリング手法を候補生成法として比較した．その結果，提案手法は全ての候補生成法において，総入出力トークン数がベースラインの 64% から 68% 程度に削減された．一方で，precision および F1 はベースラインと同等以上であり，また recall の低下は最大でも 0.05 に留まった．これらより，提案手法は，矛盾検知性能をほぼ保ちつつ API コストを削減する有効な手段であると考えられる．

7）https://github.com/openai/tiktoken

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 (EMNLP-IJCNLP)**, pp. 2463–2473, November 2019.

[2] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. REALM: retrieval-augmented language model pre-training. In **Proceedings of the 37th International Conference on Machine Learning**, ICML'20, 2020.

[3] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In **Advances in Neural Information Processing Systems**, Vol. 33, pp. 9459–9474, 2020.

[4] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-Augmented Generation for Large Language Models: A Survey. arXiv Preprint arXiv:2312.10997, 2024.

[5] Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. Evaluating the factual consistency of abstractive text summarization. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 9332–9346, Online, November 2020. Association for Computational Linguistics.

[6] Jie Ouyang, Tingyue Pan, Mingyue Cheng, Ruiran Yan, Yucong Luo, Jiaying Lin, and Qi Liu. HoH: A dynamic benchmark for evaluating the impact of outdated information on retrieval-augmented generation. In **Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 6036–6063, July 2025.

[7] Cheng Jiayang, Chunkit Chan, Qianqian Zhuang, Lin Qiu, Tianhang Zhang, Tengxiao Liu, Yangqiu Song, Yue Zhang, Pengfei Liu, and Zheng Zhang. ECON: On the Detection and Resolution of Evidence Conflicts. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, **Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing**, pp. 7816–7844. Association for Computational Linguistics, November 2024.

[8] OpenAI. Pricing - openai api. https://platform.openai.com/docs/pricing. Accessed: 2026-01-09.

[9] Sarwosri, Umi Laili Yuhana, and Siti Rochimah. Conflict detection of functional requirements based on clustering and rule-based system. **IEEE Access**, Vol. 12, pp. 174330–174342, 2024.

[10] Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. Knowledge conflicts for llms: A survey. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, **Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024**, pp. 8541–8565. Association for Computational Linguistics, 2024.

[11] Arie Cattan, Alon Jacovi, Ori Ram, Jonathan Herzig, Roee Aharoni, Sasha Goldshtein, Eran Ofek, Idan Szpektor, and Avi Caciularu. DRAGged into Conflicts: Detecting and Addressing Conflicting Sources in Search-Augmented LLMs. arXiv:2506.08500, 2025.

[12] Vignesh Gokul, Srikanth Tenneti, and Alwarappan Nakkiran. Contradiction Detection in RAG Systems: Evaluating LLMs as Context Validators for Improved Information Consistency. arXiv:2504.00180, 2025.

[13] Han Wang, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. Retrieval-Augmented Generation with Conflicting Evidence. arXiv:2504.13079, 2025.

[14] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In Lluís Màrquez, Chris Callison-Burch, and Jian Su, editors, **Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing**, pp. 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[15] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, **Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)**, pp. 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[17] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. In **International Conference on Learning Representations (ICLR)**, 2021. Poster.

[18] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. airXiv Preprint airXiv:1907.11692, 2019.

[19] Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. Summac: Re-visiting nli-based models for inconsistency detection in summarization. **Transactions of the Association for Computational Linguistics**, Vol. 10, pp. 163–177, 2022.

[20] John Glover, Federico Fancellu, Vasudevan Jagannathan, Matthew R. Gormley, and Thomas Schaaf. Revisiting text decomposition methods for nli-based factuality scoring of summaries. In **Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)**, pp. 97–105, 2022.

[21] Tal Schuster, Sihao Chen, Senaka Buthpitiya, Alex Fabrikant, and Donald Metzler. Stretching sentence-pair nli models to reason over long documents and clusters. In **Findings of the Association for Computational Linguistics: EMNLP 2022**, pp. 394–412, 2022.