ハルシネーションから学ぶ：内部表現への介入によるハルシネーション抑制

門谷宙　西田光甫　西田京介
NTT 株式会社　人間情報研究所
{sora.kadotani,kosuke.nishida,kyosuke.nishida}@ntt.com

概要

大規模言語モデル (LLM) は，事実と異なる情報を生成すること (ハルシネーション) がある．近年，推論時に嘘つき LLM (anti-expert) の出力確率をペナルティとして用いることでハルシネーションを抑制できることが示されている．anti-expert 手法は，ハルシネーション抑制の最高性能を達成しているが，2 つの LLM を同時に動かす必要があるため推論コストが高い．そこで，本研究では in-model anti-expert という軽量な anti-expert 手法を提案する．提案手法は，LLM の内部表現を事実性が向上する方向に編集することで，モデル単体でのハルシネーション抑制を実現する．評価実験の結果，提案手法は anti-expert 以外の既存手法を上回る精度を達成した．また，提案手法は従来の anti-expert 手法の GPU メモリ使用量を 2.2 倍から 1.4 倍，推論時間を 1.9 倍から 1.2 倍に改善することを確認した．

1　はじめに

大規模言語モデル (LLM) [1, 2] は，驚くべき能力を示している．その一方で，LLM は尤もらしいが事実と異なる情報を生成することがあり，この問題はハルシネーション [3, 4] と呼ばれる．ハルシネーションは，アプリケーションの信頼性を著しく低下されるため，検知や抑制で防ぐことが重要である．

有望なハルシネーション抑制手法の一つに，anti-expert 手法 [5] がある．事実性を直接向上させる fine-tuning は LLM に知識の有無に関わらず回答することを強制するため，逆にハルシネーションを増加させる [6]．anti-expert 手法では，対象の LLM (*i.e.* base LLM) を嘘が含まれるデータで fine-tuning して嘘つき LLM (*i.e.* anti-expert LLM) を構築する．そして，推論時に base LLM の出力確率から anti-expert LLM の出力確率をペナルティとして差し引くことで事実性が向上した出力確率を得る．anti-expert 手法は，ハルシネーション抑制の最高性能だが，同時に 2 つの LLM を動かすため推論コストが高い．

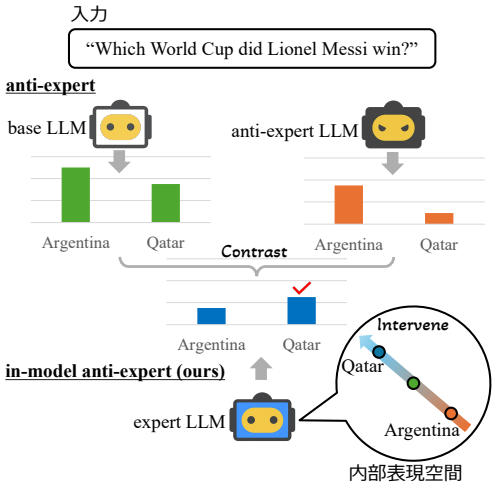


図 1　anti-expert (上部) と in-model anti-expert (下部)．提案手法は，出力確率を直接操作するのではなく，base LLM の内部表現を事実性が向上する方向に編集する．

本研究では，in-model anti-expert (IMAE) という軽量な anti-expert を提案する．図 1 に anti-expert と IMAE の比較を示す．anti-expert は出力確率を直接操作するが，IMAE は内部表現に介入して事実性を改善する．内部表現への介入は，追加の LLM を動かす必要性をなくし，モデル単体でのハルシネーション抑制を可能にする．しかし，本手法の実現には，LLM が嘘を生成できることに加え，内部表現空間において事実性を高める方向を同定できることが前提となり，技術的課題が存在する．この課題に対して，我々は LLM に新たなモジュールを追加し，3 つの生成モードを搭載することで解決する．

Truthfulqa [7] を用いた評価実験において，IMAE は主要な既存手法を上回り，anti-expert に次ぐ精度を達成した．また，IMAE は従来の anti-expert の GPU メモリ使用量を 2.2 倍から 1.4 倍，推論時間を 1.9 倍から 1.2 倍に改善することを確認した．

2 背景

本節では，anti-expert 手法について説明した後，ほかのハルシネーション抑制手法について述べる．

2.1 前提知識：anti-expert

ハルシネーション抑制の大きな課題は，事実データを用いた fine-tuning は逆に LLM の事実性を低下させることである [8]．従来の fine-tuning は，LLM の知識で解けない質問に対しても回答を強制し，事実を捏造するハルシネーションを増加させる [6]．

そこで，有害性抑制の分野で提案された手法 [9] を拡張した induce-then-contrast decoding (ICD) [5] が提案された．ICD は，LLM が嘘をつく能力を容易に獲得できる点に着目した手法である．ICD は，base LLM を ChatGPT で生成した嘘を含むデータで fine-tuning して anti-expert LLM を構築する．そして，推論時に base LLM の出力確率から anti-expert LLM の出力確率をペナルティとして差し引くことで事実性が向上した出力確率 $p_{\text{expert}}(\cdot)$ を得る．

$$p_{\text{expert}}(x_i|x_{<i}) = \text{softmax}(\beta \log p_{\text{base}}(x_i|x_{<i}) - \log p_{\text{anti}}(x_i|x_{<i})) \quad (1)$$

$p_{\text{base}}(\cdot)$ と $p_{\text{anti}}(\cdot)$ はそれぞれ base LLM と anti-expert LLM の出力確率を表す．また，x_i は i 番目のトークン，$x_{<i}$ は i 番目より前のすべてのトークンを表す．β はペナルティの大きさを決めるハイパーパラメータである．anti-expert の出力確率が大きいトークンほど，ペナルティが大きくなる．

すべてのトークンをペナルティの対象にすると生成能力が低下することが分かっている [10]．ICD では一部のトークン $\mathcal{V}_{\text{valid}}$ にのみペナルティを与える．

$$\mathcal{V}_{\text{valid}} = \{x_i \in \mathcal{V} : \text{logit}_{\text{base}}(x_i|x_{<i}) \geq \delta \max_i (\max_\omega \text{logit}_{\text{base}}(\omega|x_{<i}))\}$$

$\text{logit}_{\text{base}}(\cdot)$ は base LLM のロジット，$\mathcal{V}$ は語彙集合を表す．δ はペナルティの対象とするトークンの制限度合いを決めるハイパーパラメータである．

ICD は TruthfulQA において最高性能を達成しており，ICD を適用した Llama2 (7B) [11] は GPT-4 [1] の精度を上回っている．しかし，ICD を含め既存の anti-epert 手法は推論コストが高い．ICD は，2 つの LLM を同時に動かすため，2.2 倍の GPU メモリ使用量がかかる．また，毎時刻出力確率を操作する処理が発生するため，1.9 倍の推論時間がかかる．

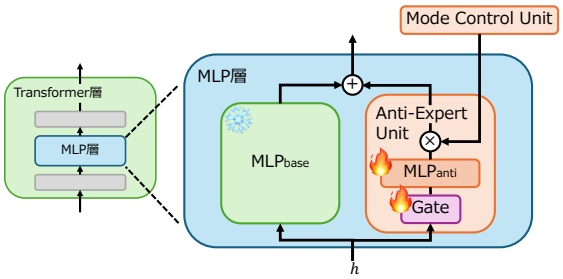


図 2 IMAE のモデル構造．

2.2 関連研究

LLM におけるハルシネーション ハルシネーション [4, 12] とは，LLM がユーザ入力 [13]，以前の出力 [14]，実世界の事実 [15, 16] のいずれかと矛盾する情報を出力することである．本研究では，深刻な問題を引き起こす危険性がある実世界の事実と矛盾するハルシネーション [17] を対象とする．

ハルシネーション抑制 主なハルシネーション抑制手法として，デコーディングアルゴリズムを改善する手法 [18, 19] が存在する．これらの手法は，軽量だが学習を伴わないため精度向上には限界がある．また，LLM に知識境界を認識させる fine-tuning 手法 [8] は，高精度だがモデルが過度に保守的な振る舞いをする問題がある．我々と同様に内部表現を編集する手法 [20] も存在する．しかし，この手法はドメイン内データでは有効であるが，正答と誤答のペアデータが必要でデータ準備が難しい．

3 提案手法：in-model anti-expert

IMAE は，base LLM の内部に anti-expert の役割を担うモジュールを追加する．そして，base LLM の内部表現を事実性が向上する方向に編集する．

3.1 モデル構造

図 2 に IMAE のモデル構造を示す．この構造はパラレルアダプタ [21] に基づいており，anti-expert unit を base LLM の各 MLP 層と並列に追加する．anti-expert unit は gate と MLP 層で構成される．また，mode control unit も追加する．MLP_{base} と MLP_{anti} は，base LLM と anti-expert unit の MLP 層を表す．

Anti-Expert Unit gate は anti-expert unit の出力をどれだけ考慮するかを決める $\alpha \in \mathbb{R}$ を計算する．

$$\alpha = \text{softmax}(Wh + b)_0 \in \mathbb{R}$$

$h \in \mathbb{R}^d$ は MLP 層への入力ベクトルを表す．$W \in \mathbb{R}^{2\times d}$ と $b \in \mathbb{R}^2$ は学習可能なパラメータである．

gate は $\mathrm{MLP}_{\mathrm{anti}}$ への入力ベクトルを出力する．

$$y = \mathrm{MLP}_{\mathrm{anti}}(\alpha h) \in \mathbb{R}^d$$

$\mathrm{MLP}_{\mathrm{base}}$ は言語理解・生成能力を保持しているため，$\mathrm{MLP}_{\mathrm{anti}}$ には嘘をつくための情報だけを埋め込めばよく，$\mathrm{MLP}_{\mathrm{base}}$ は $\mathrm{MLP}_{\mathrm{anti}}$ よりも小さくできる．

Mode Control Unit　IMAE には 3 つのモードがあり，mode control unit は各モードに対応するスカラ値 $\sigma \in \{-1, 0, 1\}$ を出力する．$\sigma = 1$ のとき事実性が低下したテキストを生成 (anti-expert モード)，$\sigma = 0$ のとき base LLM と同じテキストを生成 (base モード)，$\sigma = -1$ のとき事実性が向上したテキストを生成 (expert モード) する．推論時は expert モードで生成する．MLP 層の出力は以下のように計算される．

$$\mathrm{MLP}(h) = \mathrm{MLP}_{\mathrm{base}}(h) + \sigma y$$

3.2　損失関数

IMAE の fine-tuning では，base LLM を凍結して anti-expert unit のパラメータのみを調整する．$\mathrm{MLP}_{\mathrm{anti}}$ の出力の逆ベクトルが，$\mathrm{MLP}_{\mathrm{base}}$ の出力ベクトルの事実性を高める方向を向くように調整する．ICD と同様に質問と誤答で構成された訓練データを用いる．

MLP 層の損失　訓練データを分割し，半分を anti-expert モード，もう半分を expert モードの訓練に用いる．マルチタスク学習により，anti-expert モードでは事実性が低下したテキスト，expert モードでは事実性が向上したテキストを生成するように訓練する．anti-expert モードの損失 L_{anti} には，cross-entropy 損失を用いる．expert モードの損失 L_{expert} には，事実性が向上した出力確率 $p_{\mathrm{target}}(\cdot)$ を用いた損失関数を提案する．$p_{\mathrm{target}}(\cdot)$ は式 1 と同様に計算する．そのとき，base LLM と anti-expert LLM の代わりに base モードと anti-expert モードの出力確率を用いて，ICD と同じく $\mathcal{V}_{\mathrm{valid}}$ にのみペナルティを与える．L_{expert} は以下のように計算される．

$$L_{\mathrm{expert}} = D_{\mathrm{KL}}(p_{\mathrm{target}}(x_i|x_{<i}) || p_{\mathrm{expert}}(x_i|x_{<i}))$$

この式において，$p_{\mathrm{expert}}(\cdot)$ は expert モードの出力確率を表す．D_{KL} は Kullback-Leibler 距離を表す．

gate の損失　$\mathrm{MLP}_{\mathrm{anti}}$ の損失だけでなく gate の損失 L_{gate} も用いる．L_{gate} を以下のように定義する．

$$L_{\mathrm{gate}} = \begin{cases} 1 - \alpha & \text{if } x_i \in T_{\mathrm{fact}} \\ \alpha & \text{if } x_i \notin T_{\mathrm{fact}} \end{cases}$$

T_{fact} はテキストの事実性に影響するトークンの集合である．T_{fact} の特定方法は §3.3 で説明する．α は T_{fact} を生成するときに大きくなり，それ以外のトークンを生成するときに小さくなる．その結果，anti-expert unit の出力は，事実性に影響するトークンを生成するときにより考慮される．全体の損失は，$L = L_{\mathrm{anti}} + L_{\mathrm{expert}} + L_{\mathrm{gate}}$ と定義する．

3.3　トークンフィルタリング

$\mathrm{MLP}_{\mathrm{anti}}$ の訓練にすべてのトークンを用いると，スタイルのような事実性に無関係な情報も埋め込まれる．そこで，事実性に影響するトークン T_{fact} を特定し，T_{fact} のみを L_{anti} と L_{expert} の計算に用いる．

ICD と同様に anti-expert LLM を構築し，base LLM と anti-expert LLM を用いて T_{fact} を特定する．

$$T_{\mathrm{fact}} = \{x_i \in T : \log p_{\mathrm{anti}}(x_i|x_{<i}) - \log p_{\mathrm{base}}(x_i|x_{<i}) \geq \gamma\}$$

γ は制限度合いのハイパーパラメータである．

4　実験

提案手法を英語の QA タスクで評価する．Zhang ら [5] の実験設定に従い，既存手法と比較する．

4.1　設定

ベンチマーク　訓練データとして HaluEval [22] を用いる．評価データには TruthfulQA [7] を用いて，多肢選択形式の評価指標 MC1/2/3 で評価する．MC1 は最尤選択肢が正解であるかを評価する．MC2 は正解選択肢の合計尤度と不正解選択肢の合計尤度の差を測る．MC3 は正解選択肢の最小尤度が不正解選択肢の最大尤度を上回るかを評価する．MC1 は，最も一般的な greedy decoding の設定に近いため，最も重要な評価指標と位置付けられる．

比較手法　base LLM には，Llama2-7B-Chat [11] を用いる．ICD に加えて，主要な既存手法である ITI [18]，DoLa [19]，CD [10] と比較する．ITI は attention を用いて内部表現を編集する．DoLa は異なる層の出力確率分布を対比する．CD はパラメータサイズの異なる LLM[1] の出力確率分布を対比する．

4.2　結果と分析

IMAE は事実性を向上させるか？　表 1 に結果を示す．最も重要な評価指標である MC1 において，IMAE は ICD 以外の既存手法を上回った．

MC2/3 においては一部の手法が IMAE を上回ったが，これは各手法の性質の違いに起因すると考

1)　Llama2-7B-Chat と Llama2-13B-Chat を用いた．

表 1 TruthfulQA における評価結果.

	MC1	MC2	MC3
ベースライン	36.96	54.62	27.95
ICD (上限)	46.32	69.08	41.25
ITI	37.01	54.66	27.82
DoLa	32.97	60.84	29.50
CD	28.15	54.87	29.75
IMAE (ours)	40.02	57.12	28.96

表 2 GPU メモリ使用量 (GB) と推論時間 (ms/token).

	GPU メモリ	推論時間
ベースライン	13.2 (**1.0x**)	2.09 (**1.0x**)
ICD	28.6 (**2.2x**)	4.05 (**1.9x**)
ITI	16.2 (**1.2x**)	2.09 (**1.0x**)
DoLa	15.1 (**1.2x**)	2.21 (**1.1x**)
CD	41.0 (**3.1x**)	6.42 (**3.1x**)
IMAE (ours)	18.4 (**1.4x**)	2.60 (**1.2x**)

える．MC1 は最尤の正解選択肢のみを考慮するが，MC2/3 はすべての正解選択肢を考慮する．IMAE は，特定のトークンの出力確率を増加させる代わりにほかのトークンの出力確率を減少させる．その結果，最尤の正解選択肢の尤度は向上したが，一部の正解選択肢の尤度は低下したと考える．一方で，現在広く使われている DoLa/CD は，事前学習モデルには有効である [10, 19] が，事後学習モデルの MC1 には効果が薄いと考える．DoLa/CD は，出力確率分布を緩やかに変化させる手法であり，MC2/3 には有効である．しかし，最尤の正解選択肢の出力確率を大きく増加させる手法ではないため，MC1 への効果は限定的である．このトレードオフを考慮したとき，IMAE はすべてのスコアを向上させ，かつ MC1 で最高性能を達成した最も優れた手法であると考える．

IMAE は推論コストを改善するか？ 表 2 に各手法の推論コストを示す．IMAE は，ICD の GPU メモリ使用量を 2.2 倍から 1.4 倍，推論時間を 1.9 倍から 1.2 倍に改善した．IMAE は，ICD の推論コストの増加を ITI や DoLa と同程度にまで軽減した．

IMAE はほかのモデルサイズに対しても有効か？ Llama3.2 (3B) および Llama3.1 (8B) [23] を用いた追加実験を行った．表 3 に実験結果を示す．IMAE は，両モデルのすべてのスコアを一貫して向上させた．この結果は，IMAE がより小規模なモデルに対しても有効であることを示している．

表 3 Llama3 系列モデルの評価結果.

	MC1	MC2	MC3
Llama3.2-3B-Instruct	35.25	54.75	27.44
+ ICD	40.13	73.67	44.67
+ IMAE (ours)	38.19	59.36	32.10
Llama3.1-8B-Instruct	40.88	59.90	31.35
+ ICD	44.80	79.03	50.99
+ IMAE (ours)	43.08	63.62	36.01

5 おわりに

anti-expert 手法は高精度なハルシネーション抑制手法であり，正答と誤答のペアデータではなく誤答データのみで訓練が可能な点でデータ準備が容易である．本研究では，従来手法の推論コストが大きいという課題を解決し，低コストなハルシネーション抑制手法を提案した．本研究の貢献を以下に示す．

本研究の独自性 本研究では，複数の LLM を用いて実現されてきた anti-expert 手法を，単一モデル内で実現した．提案手法は，出力確率を直接操作するのではなく，内部表現に介入して事実性を向上させる新たな枠組みを提示した．さらに，LLM に 3 つの生成モードを導入することで，嘘を生成する能力と内部表現空間で事実性を高める方向を同定する能力を同時に獲得させるという技術的課題を解決した．本研究は，ハルシネーション抑制に対して有効かつスケーラブルな新しい解決策を示した．

本研究の重要性 現在主流のハルシネーション対策は RAG [24] であり，その派生手法 [25, 26] が世界的に活発に研究されている．近年では，ハルシネーションを原因ごとに分類し，各分類に応じた対策を講じる必要性が指摘されている [27, 28].

ハルシネーションの主な原因としては，知識不足 [29] と知識の想起失敗 [30] が挙げられる．RAG は前者には有効であるが後者には有効でない．これに対して，提案手法は前者には有効でないが後者には有効である．本研究は RAG 研究の系譜とは直交するものであるが，提案手法は RAG ベースの手法と併用することでさらなる精度向上が期待できる．本研究は，現在主流の手法では十分に対処されていない知識の想起失敗に起因するハルシネーションを対象とし，原因別対策という観点からハルシネーション抑制研究を補完する点で重要である．

参考文献

[1] OpenAI. Gpt-4 technical report. In **arXiv:2303.08774**, 2024.

[2] Nvidia. Nemotron-4 340b technical report. In **arXiv:2406.11704**, 2024.

[3] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. **ACM Comput. Surv.**, Vol. 55, No. 12, 2023.

[4] Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. Siren's song in the ai ocean: A survey on hallucination in large language models. In **arXiv:2309.01219**, 2023.

[5] Yue Zhang, Leyang Cui, V. W., and Shuming Shi. Alleviating hallucinations of large language models through induced hallucinations. In **NAACL Findings**, pp. 8218–8232, 2025.

[6] Yuqing Yang, Ethan Chern, Xipeng Qiu, Graham Neubig, and Pengfei Liu. Alignment for honesty. In **NeurIPS**, 2024.

[7] Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In **ACL**, pp. 3214–3252, 2022.

[8] Hanning Zhang, Shizhe Diao, Yong Lin, Yi Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. R-tuning: Instructing large language models to say 'I don't know'. In **NAACL-HLT**, pp. 7113–7139, 2024.

[9] Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. DExperts: Decoding-time controlled text generation with experts and anti-experts. In **ACL-IJCNLP**, pp. 6691–6706, 2021.

[10] Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive decoding: Open-ended text generation as optimization. In **ACL**, pp. 12286–12312, 2023.

[11] Meta. Llama 2: Open foundation and fine-tuned chat models. In **arXiv:2307.09288**, 2023.

[12] Nouha Dziri, Sivan Milton, Mo Yu, Osmar Zaiane, and Siva Reddy. On the origin of hallucinations in conversational models: Is it the datasets or the models? In **NAACL-HLT**, pp. 5271–5285, 2022.

[13] David Dale, Elena Voita, Loic Barrault, and Marta R. Costa-jussà. Detecting and mitigating hallucinations in machine translation: Model internal workings alone do well, sentence similarity Even better. In **ACL**, pp. 36–50, 2023.

[14] David Wan, Shiyue Zhang, and Mohit Bansal. HistAlign: Improving context dependency in language generation by aligning with history. In **EMNLP**, pp. 2941–2960, 2023.

[15] Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity. In **IJCNLP-AACL**, pp. 675–718, 2023.

[16] Xuming Hu, Junzhe Chen, Xiaochuan Li, Yufei Guo, Lijie Wen, Philip S. Yu, and Zhijiang Guo. Do large language models know about facts? In **arXiv:2310.05177**, 2023.

[17] Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Med-HALT: Medical domain hallucination test for large language models. In **CoNLL**, pp. 314–334, 2023.

[18] Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In **NeurIPS**, 2023.

[19] Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models. In **ICLR**, 2024.

[20] Shaolei Zhang, Tian Yu, and Yang Feng. TruthX: Alleviating hallucinations by editing large language models in truthful space. In **ACL**, pp. 8908–8949, 2024.

[21] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. In **ICLR**, 2022.

[22] Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In **EMNLP**, pp. 6449–6464, 2023.

[23] Meta. The llama 3 herd of models. In **arXiv:2407.21783**, 2024.

[24] 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 **ICLR**, 2020.

[25] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In **ICLR**, 2024.

[26] Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. The power of noise: Redefining retrieval for rag systems. In **SIGIR**, 2024.

[27] Yejin Bang, Ziwei Ji, Alan Schelten, Anthony Hartshorn, Tara Fowler, Cheng Zhang, Nicola Cancedda, and Pascale Fung. HalluLens: LLM hallucination benchmark. In **ACL**, pp. 24128–24156, 2025.

[28] Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and Edwin Zhang. Why language models hallucinate. In **arXiv:2509.04664**, 2025.

[29] Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In **ACL**, pp. 9802–9822, 2023.

[30] Shen Zheng, Jie Huang, and Kevin Chen-Chuan Chang. Why does chatgpt fall short in providing truthful answers? In **ICBINB@NeurIPS**, 2023.

[31] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In **ICLR**, 2019.

表 4　IMAE のハイパーパラメータ．

Configuration	Value
エポック数	5
バッチサイズ	24
学習率	5×10^{-5}
$\mathrm{MLP}_{\mathrm{anti}}$ の中間層の次元数	4096
$p_{\mathrm{expert}}(\cdot)$ のペナルティの強さ β	2.0
T_{fact} の制限度合い γ	1.0
v_{valid} の制限度合い δ	1.0

A　実験設定詳細

訓練データには，HaluEval [22] から 10k 件の質問とそれに対する誤答で構成されたデータを用いた．Halueval の誤答は，ChatGPT に嘘の回答を生成するように指示することで，擬似的に生成している．評価データには，ハルシネーション抑制の分野で広く用いられている TruthfulQA [7] を用いた．TruthfulQA は，LLM がハルシネーションを起こしやすい 817 件の 質問で構成されており，医療・法律・政治など 38 分野を網羅している．

実験には，NVIDIA RTX 6000 (24GB) を 8 枚用いた．表 4 に IMAE のハイパーパラメータを示す．最適化アルゴリズムには，AdamW [31] を用いた．

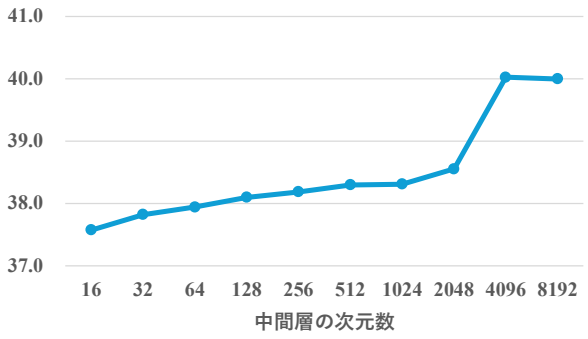
B　アブレーション

提案手法のどの構成要素が精度向上に寄与するか？　表 5 は，TruthfulQA における Llama2-7B-Chat のアブレーション実験結果である．損失の各項について，anti-expert モードの損失 L_{anti}，expert モードの損失 L_{expert}，gate の損失 L_{gate} のいずれを取り除いても，精度が著しく低下した．この結果は，損失のすべての要素が重要であることを示している．また，gate やトークンフィルタリングは，さらなる精度向上に貢献することも確認した．

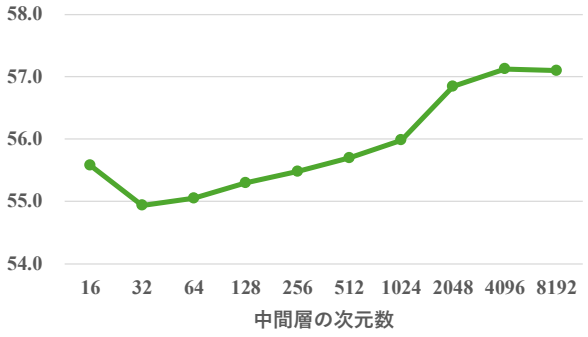
anti-expert unit の MLP 層の中間層の次元数は精度に影響するか？　図 3 に $\mathrm{MLP}_{\mathrm{anti}}$ の中間層の次元数と MC1/2/3 の関係を示す．$\mathrm{MLP}_{\mathrm{base}}$ の中間層の次元数は 11008 である．MC1/2/3 は，$\mathrm{MLP}_{\mathrm{anti}}$ の次元数が 4096 までは向上する傾向があったが，4096 以上はいずれも向上しなかった．この結果は，ある一定の値までは $\mathrm{MLP}_{\mathrm{anti}}$ の中間層の次元数が増加するにつれて事実性が向上することを示している．

表 5　TruthfulQA におけるアブレーション実験結果．

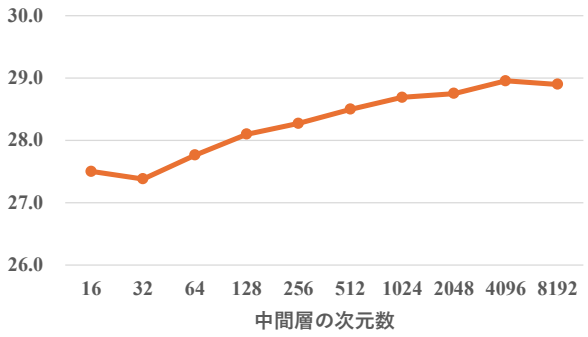
	MC1	MC2	MC3
ベースライン	36.96	54.62	27.95
IMAE	40.02	57.12	28.96
w/o gate	39.05	55.96	29.07
w/o L_{anti}	37.09	54.15	27.68
w/o L_{expert}	36.35	54.08	26.64
w/o L_{gate}	37.33	54.28	27.24
w/o トークンフィルタリング	38.80	55.46	28.43



(a) MC1



(b) MC2



(c) MC3

図 3　TruthfulQA における $\mathrm{MLP}_{\mathrm{anti}}$ の中間層の次元数ごとの精度変化．base LLM は Llama2-7B-Chat である．