大規模言語モデルはデータ漏洩を隠蔽できるのか

高橋侑成 [1]　Youmi Ma[1]　金子正弘 [2,1]　岡崎直観 [1,3,4]
[1] 東京科学大学　[2] MBZUAI　[3] 産業技術総合研究所　[4] NII LLMC
{yukinari.takahashi@nlp.,ma.y@,okazaki@}comp.isct.ac.jp
masahiro.kaneko@mbzuai.ac.ae

概要

特定の文章が機械学習モデルの学習に漏洩しているかを推論する手法として，メンバーシップ推論攻撃（MIA）がある．大規模言語モデル（LLM）に対する MIA では，文字の並びなどの表層情報を利用するが，LLM はテキストをそのまま記憶しているとは限らない．本稿では，テキストの忘却と質問応答タスクの学習を同時に行うことで，LLM にテキストの文字の並びを忘却させながら，その知識を保持できることを報告する．すなわち，MIA の成功率を低減しながら，関連知識についての質問応答の性能を維持できる．この知見は，たとえテキストが MIA で推論されなくても，その知識を LLM が隠蔽できている可能性があると警鐘を鳴らすものである．

1　はじめに

大規模言語モデル（Large Language Model; LLM）の学習データは膨大で，その中身を全て確認することは困難である．しかし，学習データのフィルタリングが不十分であると，個人情報や著作物，ベンチマークデータなど，公にしたくないテキストが学習データに混ざり，**データ漏洩**のリスクが高まる．データ漏洩により，LLM による個人情報や著作物の再生成 [1, 2]，LLM の性能の過大評価 [3, 4] などの問題が引き起こされる．

与えられたテキストが LLM の学習データに含まれているかどうかを推定する手段として，メンバーシップ推論攻撃（**M**embership **I**nference **A**ttacks; **MIA**）[5] がある．MIA は典型的に，検出対象となるテキストが漏洩している可能性を定量化したスコアを計算し，そのスコアを閾値と比較することで，学習データに含まれているか否か分類する [6]．MIA の従来手法は，文字の並びなどテキストの表層的な一致度に依存している [7, 8]．ところが，LLM はテキストをその文字の並びの通りに記憶しているとは限らない [9]．そのため，表層的にはモデルが対象テキストを「忘却」していても，そのテキストの情報を「保持」できる可能性がある．

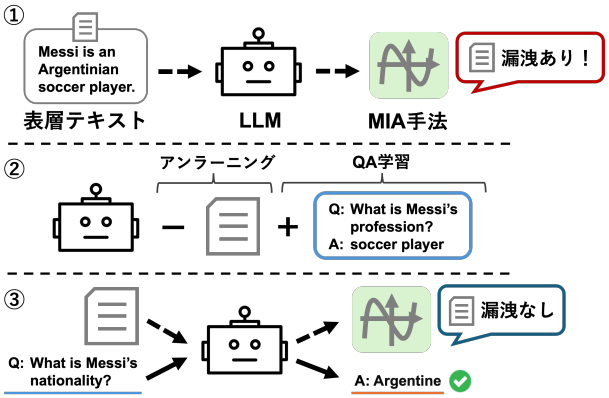

図 1　提案手法の概略．中段は提案手法による学習を表す．

本稿では，図 1 のように MIA を回避しながら，LLM が対象テキストの知識を必要とするタスクに正解できてしまうことを報告する．本研究では，対象タスクを質問応答タスク（以降 QA タスクと呼ぶ）とし，表層テキストを LLM に忘却させるアンラーニング手法 [10] と QA タスクの性能維持のための対数尤度最大化を組み合わせたマルチタスク学習を提案する．提案手法の有効性を検証するために，背景情報付き質問応答データセット HotpotQA [11] を用い，背景情報内にある質問の根拠文を MIA から隠蔽しつつ，その根拠文に関する QA タスクの性能を維持できるか，実験を行う．その結果，提案手法は全ての実験設定において，根拠文を MIA で推論されないようにしつつ，ベースモデルよりも高い正解率で根拠文に関する質問を回答できた．ゆえに，従来のデータ漏洩検出手法はテキストおける文字の並びなどの表層情報の検出に留まっており，テキストで記述されている知識など，より抽象化・一般化された情報を検出するには，新しい手法が必要である．

2 提案手法

本研究では，テキストの情報が無ければ答えられない質問に関して，LLM の回答能力を維持しながら，そのテキストの学習データ中の存在を MIA で見破られないようにすることは可能か，検証したい．このため，MIA によるテキストの検出を回避するアンラーニング [10] と，QA タスク性能を維持する対数尤度最大化のマルチタスク学習を提案する．テキストの集合を C，C 内の複数のテキストに関連する質問 q と応答 a の集合を $H = \{(q,a)\}$ とする．データセット D を (C,H) とする．LLM が C を隠蔽し，H 内の質問から正しい回答を出力可能にする．以下のようにマルチタスク学習の損失関数 $\mathscr{L}_{\mathrm{m}}$ を定義する．

$$\mathscr{L}_{\mathrm{m}} = \mathscr{L}_{\mathrm{u}}(C) + \mathscr{L}_{\mathrm{t}}(H) \tag{1}$$

ここで，$\mathscr{L}_{\mathrm{u}}$ と $\mathscr{L}_{\mathrm{t}}$ はそれぞれアンラーニングと QA の損失関数である．

まず，アンラーニングの損失関数について説明する．先行研究 [10] に倣い，尤度を下げたいテキストの集合（以降忘却集合と呼ぶ）と尤度を維持したいテキストの集合（以降非忘却集合と呼ぶ）の両方を利用する．LLM がテキスト $c \in C$ に対して計算した負の対数尤度を，損失関数 $\ell(c)$ とする．また，C を忘却集合として使用し，それと交わらない非忘却集合 C_{r} を用意し，損失関数を以下のように定める．

$$\mathscr{L}_{\mathrm{u}}(C) = \mathscr{L}_{\mathrm{r}}(C_{\mathrm{r}}) - \mathscr{L}_{\mathrm{f}}(C), \tag{2}$$

$$\mathscr{L}_{\mathrm{r}}(C_{\mathrm{r}}) = \frac{1}{|C_{\mathrm{r}}|} \sum_{c_{\mathrm{r}} \in C_{\mathrm{r}}} \ell(c_{\mathrm{r}}), \tag{3}$$

$$\mathscr{L}_{\mathrm{f}}(C) = \frac{|C|}{\sum_{c \in C} \frac{1}{\min(\ell(c), \gamma)}} \tag{4}$$

ここで，γ は上限値を表すハイパーパラメータであり，$\mathscr{L}_{\mathrm{r}}$，$\mathscr{L}_{\mathrm{f}}$ はそれぞれ非忘却集合と忘却集合の損失関数である．$\mathscr{L}_{\mathrm{f}}$（式 4）では，忘却集合内の各テキスト c に対しての損失関数の値を γ で切り捨て，過剰な勾配上昇により無意味な文字列が出力されるなどのモデル破壊 [12] を防ぐ．さらに $\mathscr{L}_{\mathrm{f}}$ では各テキストに対する損失関数に調和平均を適用し，忘却が進められていないテキストによる影響が大きくなるように調整することで，全てのテキストをバランス良くアンラーニングするように促す．

次に，QA の損失関数について説明する．本手法では学習の効率と性能の向上のために，応答部分の損失関数 $\ell(a \mid q)$ のみを使用して QA を学習する．

$$\mathscr{L}_{t}(H) = \sum_{(q,a) \in H} \ell(a \mid q) \tag{5}$$

3 実験

3.1 実験設定

検証対象　タスク性能の維持が難しくなる順に，以下の 3 つの設定で提案手法の有効性を検証する．

設定 1［$-C + C_r + H$］　評価とマルチタスク学習の両方で同じ (C,H) を用いる．これは評価時と同一の QA データで学習するため，マルチタスク学習による QA 性能の上限を確認できる．

設定 2［$-C + C_r + H'$］　マルチタスク学習を (C,H') に対して行う．ここで H' は H の質問を言い換えたものである．これにより LLM は，評価時と同一でないが，類似する質問からテキストの知識に対する回答能力を保持できるか確認する．言い換えには GPT-4o mini[1] を使用した．

設定 3［$-C + C_r + H''$］　マルチタスク学習を (C,H'') に対して行う．ここで H'' はテキスト C から新たに生成した QA データである．これにより，学習時とは異なる問題と正答に対して，保持しようとした知識や回答能力を活用できるか検証する．QA データの生成には，精度が高いことが期待できる GPT-4o[2] を用いた．

また，ベースラインとして，提案手法と以下二つの設定を比較する．

テキストのアンラーニングのみ［$-C + C_r$］
$\mathscr{L}_u(C)$ のみを最適化する．テキストを LLM に忘却させるため，QA の性能が最も低くなると想定している．

アンラーニングと言い換えの学習［$-C + C'$］　$\mathscr{L}_u$ の非忘却集合 C_r を C の言い換えである C' で置換し，最適化する．QA に必要な知識を C' で学習するため，アンラーニングのみの設定より高い QA 性能が期待できる．

データセット　実験を行うには，テキスト C，テキストについての質問応答 H の 2 つ組のデータセットが必要となる．以上の要件を満たす，背景情報付き質問応答データセットの HotpotQA [11] を用いる．

1) https://openai.com/ja-JP/index/gpt-4o-mini-advancing-cost-efficient-intelligence/
2) https://openai.com/index/gpt-4o-system-card/

表 1　ベースモデルと比較した各手法の QA 性能と MIA 検出率．C は隠蔽対象の根拠文，H は隠蔽対象と関連する質問応答の組，H' は H の質問を言い換えたもの，H'' は C から生成した質問応答の組，C' は C の言い換え，C_r はアンラーニングの非忘却集合である．– はアンラーニング，+ は学習を意味する．

	HotpotQA	MIA 検出率			
	F1 ↑	LOSS ↓	Reference ↓	Min-20% ↓	Min-20%++ ↓
Pythia-6.9b	0.209	0.333	0.650	0.434	0.800
ベースライン手法					
$-C+C_r$	0.154 -26.3%	0.000 -100.0%	0.006 -99.1%	0.003 -99.3%	0.024 -97.0%
$-C+C'$	0.176 -15.8%	0.000 -100.0%	0.001 -99.8%	0.001 -99.8%	0.001 -99.9%
提案手法					
$-C+C_r+H$	0.304 +45.5%	0.000 -100.0%	0.003 -99.5%	0.001 -99.8%	0.009 -98.9%
$-C+C_r+H'$	0.283 +35.4%	0.001 -99.7%	0.004 -99.4%	0.002 -99.5%	0.020 -97.5%
$-C+C_r+H''$	0.222 +6.2%	0.000 -100.0%	0.002 -99.7%	0.000 -100.0%	0.018 -97.8%

表 2　実験に使用する各データの件数．GPT-4o mini は言い換え時に複数の根拠文を 1 文とみなす場合があるため，C' は C よりも少なくなっている．また H'' は C から質問応答を 2 組ずつ生成させた．

C	H	C'	H'	H''	C_r
55,569	23,921	55,521	23,921	111,138	23,921

HotpotQA は Wikipedia から作成されたデータセットであり，背景情報は回答に直接関連する複数の根拠文から構成され，根拠文についての知識がなければ，質問に正しく答えられないように設計されている．我々は背景情報内の根拠文を C として用いる．HotpotQA における全事例のうち，根拠文全てが Pile データセット [13] に含まれている部分を選び出し，実験対象 D とする．データの選出は，Pile 作成に使用された Wikipedia ダンプ（TensorFlow Wikipedia dataset）を用いた．また，アンラーニングに用いる非忘却集合 C_r は，D に選出されなかった部分から，$C \cap C_r = \emptyset$ かつ $|C_r| = |H|$ となるように選ぶ．全実験設定の事例数を表 2 にまとめる．

LLM　提案手法の有効性を Pythia-6.9b [14] を用いて検証する．Pythia-6.9b は公開済のリソースである Pile で学習されており，根拠文が LLM の学習に使用されたかどうかを確認できる．

MIA 手法　根拠文が LLM の学習データに含まれているか否かを推定するため，MIA スコアを計算する．根拠文 c の MIA スコアが閾値 τ_{MIA} より低い場合，c が LLM の学習データに含まれるとする．なお，τ_{MIA} の算出方法を付録 A に記載した．MIA スコアの算出に，以下 4 つの手法を用いる．

- ***LOSS***[15] は最も代表的な手法で，テキストに対する LLM の損失関数を MIA スコアとする．
- ***Reference***[1] は参照モデルの損失関数を利用して LOSS を調整したものである．参照モデルには，Michael ら [6] の実験で最も検出率が高い StableLM-Base-Alpha-3B-v2 [16] を用いる．
- ***Min-K%***[7] はテキスト内のトークン単位の対数尤度を算出し，小さい K% の負の平均をスコアとしたものである．
- ***Min-K%++***[8] はトークンの対数尤度を正規化した値のうち，小さい K% の負の平均をスコアとしたものである．

なお，Min-K%と Min-K%++では，Min-K%の論文内で最良の結果を示した $K = 20$ を採用する．

学習　学習には Transformers の Trainer[3] を使用し，学習率は 2e-4 とした．忘却集合における各事例の損失関数の値を切り捨てる基準値 γ を $\{5, 10, 15, 20\}$ の範囲で探索した結果，$\gamma = 15$ とした（付録 B）．学習データから 3,000 件乱択し，LOSS による根拠文の検出率が 0.010 未満，かつ両隣よりも検出率が低くなる最初のエポックで学習を停止した．各学習手法の停止エポック数は付録 C に掲載する．

評価　QA タスクの性能を測るため，HotpotQA 事例の質問から 5-shot プロンプトを作成し，LLM に回答させ，その回答と正答の F1 スコアを計算する．なお，プロンプトは学習に使用しなかった QA 事例から 5 つ選んで作成した．また，MIA からの隠蔽度合いを測るため，根拠文に対する各 MIA の検出率を計測する．

3.2　実験結果

各設定で学習を行った後，LLM の QA タスクでの F1 スコアと根拠文の検出率を表 1 に示す．スコア

3）https://huggingface.co/docs/transformers/ja/main_classes/trainer

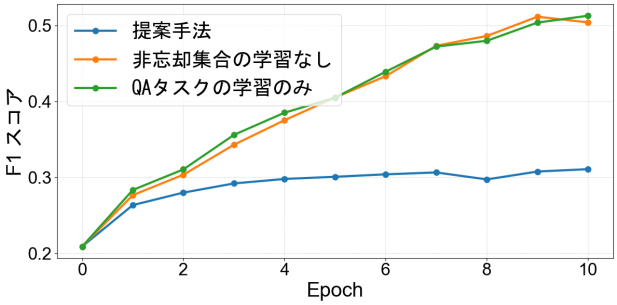

図 2　QA タスク性能の推移

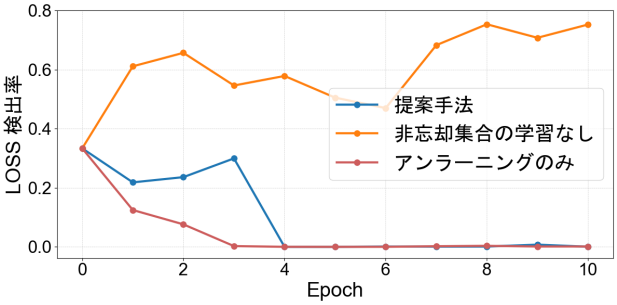

図 3　LOSS 検出率の推移

の横にはベースモデルに対する増減率を記載する．

まず，Pythia-6.9b から根拠文 C をアンラーニングすると，HotpotQA での F1 スコアが 0.055 低下した．これは，根拠文についての知識がなければ QA タスクの精度も落ちるという，HotpotQA における根拠文と質問応答の相関を裏付ける結果である．また，C の言い換え C' を非忘却集合とした設定では，C をアンラーニングする設定と比べて，QA タスクの精度が若干改善したが，ベースモデルに届かないままであった．一方，C に対しての MIA の検出率はベースモデルより大きく低下した．これにより，**MIA によるデータ漏洩の検出は，文字の並びなど表層的なレベルに留まっている**ことが示唆された．

次に，提案手法の有効性に注目する．ベースモデルと比較すると，どの設定においても QA タスクの精度が向上した．さらに全ての MIA に対して，検出性能を 0.020 以下にまで大幅に低下させることができた．ベースライン手法と比較すると，MIA の検出率は同程度であるが，QA タスクの精度が大きく向上した．したがって，テキストのアンラーニングと同時に，テキストに関するタスクを LLM に教え込むことで，**テキストを MIA に推定されないまま，その知識を保持できる**ことが示された．特に，三つの設定のうち，根拠文 C から作成した QA のセット H'' で学習を行っても，H での QA タスクの性能を回復できた．よって，**QA タスクの性能維持は，評価時と完全に一致する問題集を用いなくても可能である**ことが分かった．

4　アブレーション分析

提案手法における損失関数の各項が，学習の効能に与える影響を調べる．具体的には，QA タスクの性能に与える影響を調査するため，(1) 提案手法の設定 1（3.1 節），(2) 非忘却集合の学習を行わない設定，(3) QA タスクの学習のみ行う設定で LLM をそれぞれ 10 エポック学習し，F1 スコアの推移を図 2 に示す．同様に，MIA からの隠蔽効果に与える影響を調査するため，(1) 提案手法の設定 1（3.1 節），(2) 非忘却集合の学習を行わない設定，(3) アンラーニングのみ行う設定で LLM を学習し，LOSS の検出率の推移を図 3 に示す．なお，ここでは，学習データから 3,000 件を乱択して評価した結果を報告する．

図 2 では，提案手法では学習が進行するにつれ，QA タスクの性能が向上した．さらに，図 3 において提案手法をアンラーニングのみを行う設定と比較すると，類似する傾向が見られる．したがって，提案手法は対象文書のアンラーニングとそれに関連する QA タスクの学習の両方を実現できたと言える．

しかし，図 2 より，提案手法において非忘却集合の学習を行わない設定と比較すると，非忘却集合の学習が QA タスクの学習を阻害していることが分かる．一方で，図 3 では，非忘却集合の学習を行わないと，LOSS による検出率が下がらず，MIA から根拠文を隠蔽できない．これにより，非忘却集合の学習は MIA からテキストを隠蔽するために必要であることが示唆される．

5　おわりに

本稿では，LLM がテキストに関する知識を保持したまま，既存のデータ漏洩検出手法による検出を回避する学習手法を提案した．具体的には，テキストのアンラーニングと QA タスクの学習を同時に行うマルチタスク学習を提案した．実験では，各データ漏洩検出手法の性能を大きく下げながら，対象テキストに関する QA 性能も維持することができた．このように，既存のデータ漏洩検出手法は表層部分のみを考慮しているという脆弱性があり，より堅牢かつ強固な手法を検討する必要があろう．今後は，他の LLM での検証実験や，本手法のような隠蔽工作への対抗策などに取り組んでいきたい．

謝辞

本研究は，文部科学省補助事業「生成 AI モデルの透明性・信頼性の確保に向けた研究開発拠点形成」の支援を受けた．また，東京科学大学のスーパーコンピュータ TSUBAME4.0 を利用して実施した．

参考文献

[1] Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In **30th USENIX Security Symposium (USENIX Security 21)**, pp. 2633–2650. USENIX Association, August 2021.

[2] Antonia Karamolegkou, Jiaang Li, Li Zhou, and Anders Søgaard. Copyright violations and large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, **Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 7403–7412, Singapore, December 2023. Association for Computational Linguistics.

[3] Yonatan Oren, Nicole Meister, Niladri S. Chatterji, Faisal Ladhak, and Tatsunori Hashimoto. Proving test set contamination in black-box language models. In **The Twelfth International Conference on Learning Representations (ICLR)**, 2024.

[4] 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 Houda Bouamor, Juan Pino, and Kalika Bali, editors, **Findings of the Association for Computational Linguistics: EMNLP 2023**, pp. 10776–10787, Singapore, December 2023. Association for Computational Linguistics.

[5] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership Inference Attacks Against Machine Learning Models . In **2017 IEEE Symposium on Security and Privacy (SP)**, pp. 3–18, Los Alamitos, CA, USA, May 2017. IEEE Computer Society.

[6] Michael Duan, Anshuman Suri, Niloofar Mireshghallah, Sewon Min, Weijia Shi, Luke Zettlemoyer, Yulia Tsvetkov, Yejin Choi, David Evans, and Hannaneh Hajishirzi. Do membership inference attacks work on large language models? In **Conference on Language Modeling (COLM)**, 2024.

[7] Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. In **The Twelfth International Conference on Learning Representations (ICLR)**, 2024.

[8] Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Frank Yang, and Hai Li. Min-K%++: Improved baseline for detecting pre-training data from large language models. arXiv:2404.02936, 2024.

[9] Michael Tänzer, Sebastian Ruder, and Marek Rei. Memorisation versus generalisation in pre-trained language models. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, **Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)**, pp. 7564–7578, Dublin, Ireland, May 2022.

[10] Bo Liu, Qiang Liu, and Peter Stone. Continual learning and private unlearning. arXiv:2203.12817, 2022.

[11] 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. In **Conference on Empirical Methods in Natural Language Processing (EMNLP)**, 2018.

[12] Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. In **First Conference on Language Modeling (COLM)**, 2024.

[13] 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:2101.00027, 2020.

[14] Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. arXiv:2304.01373, 2023.

[15] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy Risk in Machine Learning: Analyzing the Connection to Overfitting . In **2018 IEEE 31st Computer Security Foundations Symposium (CSF)**, pp. 268–282, Los Alamitos, CA, USA, July 2018. IEEE Computer Society.

[16] Jonathan Tow. StableLM Alpha v2 Models. https://huggingface.co/stabilityai/stablelm-base-alpha-3b-v2.

A　MIA 閾値の決定方法

WikiMIA ベンチマーク [7] を用いて MIA の閾値を決定する．HotpotQA の根拠文の平均単語数は 22.3 であるため，それに最も近い 32 単語のデータを使用する．各学習後の LLM を用いてそれぞれの MIA のスコアを計算し，ROC 曲線を描写する．ROC 曲線上で直線 $FPR = TPR$ から $(0, 1)$ 側で最も離れている点を MIA 性能の良い点とし，その点の閾値を LLM の MIA 閾値として採用する．

B　アンラーニングにおける基準値 γ の影響

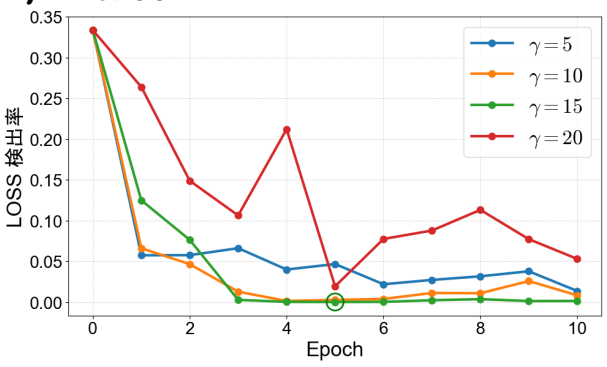

図 4　上限値 γ と LOSS 検出率の推移

各 γ の値を変化させ，アンラーニング損失関数 $\mathcal{L}_{\mathrm{u}}$（式 2）を最適化した推移を図 4 に示す．学習データのうち 3000 件乱択し，LOSS を用いて根拠文の検出率を測り，隠蔽の性能を評価する．収束が早く，検出率の推移が安定しているため，$\gamma = 15$ を採用した．また学習の停止基準に当てはまるのは 5 エポック時点（図 4 中の緑丸）である．

C　学習停止エポック数

表 3　学習手法の停止エポック

学習手法	停止エポック
アンラーニング	5
言い換えテキストの学習	5
マルチタスク学習 設定 1	4
マルチタスク学習 設定 2	6
マルチタスク学習 設定 3	4

実験で採用した停止基準を満たすエポックを表 3 に示す．

D　非忘却集合の有無による影響

提案手法の損失関数 $\mathcal{L}_m$（式 1）において，非忘却集合の損失関数 $\mathcal{L}_r$（式 3）が検出の隠蔽性能に与える影響を調査する．図 5 に $\mathcal{L}_r$ の有無による，提案手法設定 1（3.1 節）学習後の，C についての LOSS スコアの分布と閾値 τ_{LOSS} を掲載する．$\mathcal{L}_r$ の有無に関わらずスコアの値は上昇しているが，$\mathcal{L}_r$ がない場合は τ_{LOSS} の値も上昇してしまう．すなわち非忘却集合を学習しない場合は，C 以外のテキストに対しても MIA スコアが上昇してしまうため隠蔽に成功できない．対照的に，非忘却集合を学習する場合は，C 以外のテキストに対してスコアを維持しているため τ_{LOSS} の値が上昇せず，対象テキストを隠蔽することができている．以上により，MIA から特定のテキストを隠蔽するには，そのテキストについて勾配上昇するだけでなく，隠蔽対象外のテキストについての尤度を保つことが必要であることが分かった．

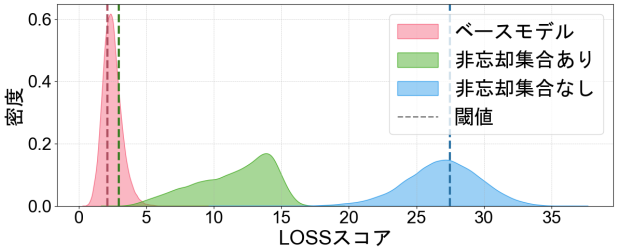

図 5　非忘却集合の有無による LOSS スコアの分布と τ_{LOSS}

E　MIA の言い換えへの対応力

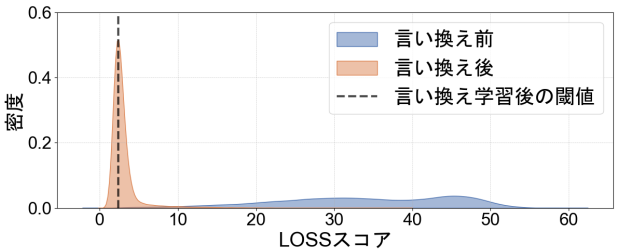

図 6　言い換え前後の LOSS スコアの分布

ベースライン手法として定義したアンラーニングと言い換えの学習を行い，言い換え前後の根拠文についての LOSS スコアの分布を，図 6 に示した．言い換えでテキストの意味は変化していないにも関わらず，言い換え前後でスコアの分布は大きく異なっている．したがって，MIA は言い換えに対応できていないことが判明した．