Quantifying Memorization and Detecting Training Data of Pre-trained Language Models using Japanese Newspaper

Shotaro Ishihara
Nikkei Inc.
Tokyo, Japan
shotaro.ishihara@nex.nikkei.com

Hiromu Takahashi
Independent Researcher
Tokyo, Japan
hiromu.takahashi56@gmail.com

Abstract

Dominant pre-trained language models (PLMs) have demonstrated the potential risk of memorizing and outputting the training data. While this concern has been discussed mainly in English, it is also practically important to focus on domain-specific PLMs. In this study, we pre-trained domain-specific GPT-2 models using a limited corpus of Japanese newspaper articles and evaluated their behavior. Experiments replicated the empirical finding that memorization of PLMs is related to the duplication in the training data, model size, and prompt length, in Japanese the same as in previous English studies. Furthermore, we attempted membership inference attacks, demonstrating that the training data can be detected even in Japanese, which is the same trend as in English. The study warns that domain-specific PLMs, sometimes trained with valuable private data, can "copy and paste" on a large scale.[1]

1 Introduction

As pre-trained language models (PLMs) have become increasingly practical, critical views on the memorization of PLMs are emerging in security and copyright (Bender et al., 2021; Bommasani et al., 2021; Weidinger et al., 2022). Prior research has indicated that neural networks have the property of unintentionally memorizing and outputting the training data (Carlini et al., 2019, 2021, 2023; Lee et al., 2023; Yu et al., 2023). In particular, Carlini et al. (2021) demonstrated that memorized personal information can be detected from GPT-2 models (Radford et al., 2019). This can lead to an invasion of privacy, reduced utility, and reduced ethical practices (Carlini et al., 2023). If there is no novelty in the generation, there would be a problem with copyright (McCoy et al., 2023; Franceschelli and Musolesi, 2023).

[1]An early version of this study was accepted for non-archival track of the Fourth Workshop on Trustworthy Natural Language Processing (Ishihara, 2024).

Research on memorization of PLMs has been intensively advanced, and empirical findings have been reported (Ishihara, 2023). Initial studies remain on the qualitative side (Carlini et al., 2021), and subsequent studies have begun to focus on quantitative evaluations. According to one of the first comprehensive quantitative studies (Carlini et al., 2023), the memorization of PLMs is strongly related to the string *duplications* in the training set, *model size*, and *prompt length*. Benchmarking of memorized string detection has also progressed, including constructing evaluation sets (Shi et al., 2024; Duarte et al., 2024; Kaneko et al., 2024; Duan et al., 2024).

These studies were conducted in English, and their reproducibility is uncertain under domain-specific conditions. Domain-specific PLMs are sometimes built on rare private corpora and have smaller pre-training corpora than general PLMs. When the data size is small, models tend to be pre-trained in multiple epochs. However, increasing the number of epochs is equivalent to string duplications, which risks increased memorization. Furthermore, security and copyright considerations become increasingly important, as the memorized contents tend to be more specific than general corpora. We, therefore, pose the following practically significant questions about domain-specific PLMs: *how much of the pre-training data is memorized*, and *is the memorized data detectable*?

This study is the first attempt to quantify the memorization of domain-specific PLMs using a limited corpus of Japanese newspaper articles. Our research objective is *to identify the memorization properties of domain-specific PLMs*. First, we developed a framework for quantifying the memorization and detecting training data of PLMs using Japanese newspaper articles (Section 3). We then pre-trained domain-specific GPT-2 models and quantified their memorization (Section 4). Furthermore, we addressed membership inference at-

Proceedings of the 17th International Natural Language Generation Conference, pages 165–179
September 23–27, 2024.

tacks (Shokri et al., 2017), which predicts whether the output string was included in the training data (Section 5).

The main findings and contributions of this paper are summarized as follows.

- **Quantification**: Japanese PLMs were demonstrated to sometimes memorize and output the training data on a large scale. Experiments reported that memorization was related to duplication, model size, and prompt length. These empirical findings, which had been reported in English, were found for the first time in Japanese.
- **Detection**: Experiments demonstrated that the training data was detected from PLMs even in Japanese. The membership inference approach suggested in English was successful with the AUC (area under the ROC curve) score of approximately 0.6. As well as the empirical findings of memorization, the more duplicates and the longer the prompt, the easier the detection was.

2 Related Work

This section reviews related work and highlights the position of this study.

2.1 Memorization of PLMs

Memorization of PLMs refers to the phenomenon of outputting fragments of the training data. Research on memorization is diverse, with various definitions and assumptions. We focus on autoregressive language models, such as the GPT family (Radford et al., 2018, 2019; Brown et al., 2020; Black et al., 2022). These are promising models and major research targets.

Definition of memorization. Many studies have adopted definitions based on partial matching of strings (Carlini et al., 2021, 2023; Kandpal et al., 2022). This definition of *eidetic memorization* assumes that memorized data are extracted by providing appropriate prompts to PLMs. Another definition of *approximate memorization* considers string fuzziness. For similarity, Lee et al. (2022) used the token agreement rate, and Ippolito et al. (2023) used BLEU.

Our study designed the first of these definitions in Japanese and reported the experimental results. Both definitions of memorization are ambiguous in languages without obvious token delimiters such as Japanese. Definitions based on the concepts of differential privacy (Jagielski et al., 2020; Nasr et al., 2021) and counterfactual memorization (Zhang et al., 2023) are beyond the scope of this study.

Issues with memorization of PLMs. Training data extraction is a security attack related to the memorization of PLMs (Ishihara, 2023). Many studies follow the pioneering work of Carlini et al. (2021). They reported that a large amount of information could be extracted by providing GPT-2 models with various prompts (generating candidates) and performing membership inference. In particular, when dealing with PLMs with sensitive domain-specific information such as clinical data, the leakage of training data can lead to major problems (Nakamura et al., 2020; Lehman et al., 2021; Jagannatha et al., 2021; Singhal et al., 2023; Yang et al., 2022). It is also necessary to discuss from the perspective of human rights, such as the right to be forgotten (Li et al., 2018; Ginart et al., 2019; Garg et al., 2020).

There has been a traditional research area for evaluating the quality of text generation, but few studies have focused on novelty (McCoy et al., 2023). Novelty in text generation is directly related to the discussion of copyright (Franceschelli and Musolesi, 2023). Lee et al. (2023) analyzed plagiarism patterns in PLMs using English domain-specific corpora.

The memorization of PLMs has also been identified as data contamination damaging the integrity of the evaluation set. Several studies have identified the inclusion of evaluation sets in the large datasets used for pre-training, which has led to unfairly high performance (Magar and Schwartz, 2022; Jacovi et al., 2023; Aiyappa et al., 2023).

Our study of quantifying memorization and performing membership inference would serve to confront these issues precisely in Japanese.

2.2 Quantifying Memorization and Detecting Training Data of PLMs

Recent studies have quantitatively evaluated memorization and related issues.

Empirical findings. As mentioned in Section 1, empirical findings in English are known that the memorization of PLMs is strongly related to the string duplications in the training set, model size, and prompt length (Carlini et al., 2021). There are supportive reports for this finding for duplication (Lee et al., 2022; Tirumala et al., 2022; Lee

et al., 2023; Ippolito et al., 2023; Kandpal et al., 2022; McCoy et al., 2023), model size (Huang et al., 2022; Kandpal et al., 2022; Lee et al., 2023; Karamolegkou et al., 2023; Ippolito et al., 2023; McCoy et al., 2023), and prompt length (Huang et al., 2022; Kandpal et al., 2022).

Evaluation sets for quantification. We describe the quantification methods used in the pioneering study (Carlini et al., 2023) and point out the potential for improvement. Owing to inference time limitations, it is impossible to evaluate memorization using all of the training data. For example, Carlini et al. (2023) targeted GPT-Neo models (Black et al., 2022) and constructed an evaluation set by sampling 50,000 samples from the Pile dataset (Gao et al., 2020) used for pre-training. Sampling and string splitting are unavoidable during the construction of the evaluation set, as shown in Figure 1. Each sampled sentence was divided into prompts of each length from 50 to 500 tokens at the beginning, with the following 50 tokens as references.

However, this splitting does not consider the importance of references. In other words, it does not consider whether references are protected subjects against security concerns. We argue that using newspaper articles can provide real-world settings in data splitting via their paywalls. Newspaper paywall restricts access to online content through a paid subscription (Myllylahti, 2016). Online news services with paid subscription plans often publish newspaper articles only at the beginning, with the rest of the text available only to their members. This system creates a real-world setting in which there is a *private part* following the *public part* as illustrated in Figure 2. Using private parts as references can achieve the splitting in which publishers hide important information that they want to preserve.

Newspaper paywalls are often discussed in the literature tied to journalism. For example, Kim et al. (2020) examined the impact of newspaper paywalls on daily page views and differences among publishers. Several other studies were conducted in the context of publishers' digital strategies (Myllylahti, 2014; Carson, 2015; Sjøvaag, 2016).

Evaluation sets for training data detection. To evaluate the detection of memorized training data from PLMs, it is necessary to have data that is guaranteed not to have been used for pre-training. A promising approach is to use new texts generated after constructing PLMs. Shi et al. (2024) constructed a dataset based on the creation date of the Wikipedia articles. Duarte et al. (2024) developed a dataset from the publication years of 165 books.

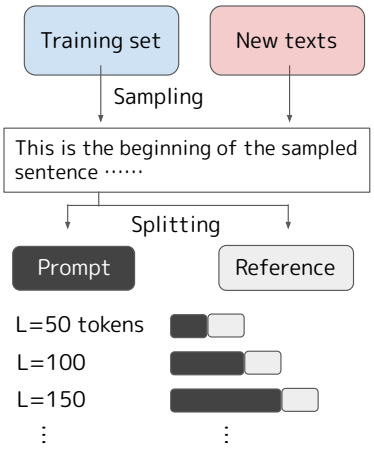

Figure 1: The existing method for constructing an evaluation set for quantifying memorization and detecting training data. This procedure requires sampling data from the training set used to pre-train and splitting the text into prompts and references. Positive examples are created from training data and negative examples from new text that are guaranteed not to be training data.

Along with evaluation sets, detection methods have been explored. For example, Shi et al. (2024) proposed Min-k% Prob, which extracts k % tokens with high log-likelihood and uses the average log-likelihood for detection. Min-k% Prob is regarded as one of the current prevailing methods (Kaneko et al., 2024; Zhang et al., 2024; Meeus et al., 2024). Kaneko et al. (2024) introduced SaMIA, which generates multiple candidates and calculates the average of the ROUGE-1 (Lin, 2004) without using the output of likelihood. The AUC score and TPR@10%FPR (True Positive Rate when False Positive Rate is 10 %) are used as the metrics (Mattern et al., 2023; Shi et al., 2024; Kaneko et al., 2024). Note that Carlini et al. (2022) recommended reporting TPR when FPR is low in membership inference assessments.

We use Japanese newspaper articles to construct the evaluation set and perform the existing detection method. Newspaper articles are generated daily, ensuring data is not used for pre-training. Given the widespread use of newspaper articles in many languages, our proposal has the appeal of high versatility in low-resource languages.

3 Problem Statement & Methodology

This section explains the problem addressed in this study and the methodology (Figure 2). We use a

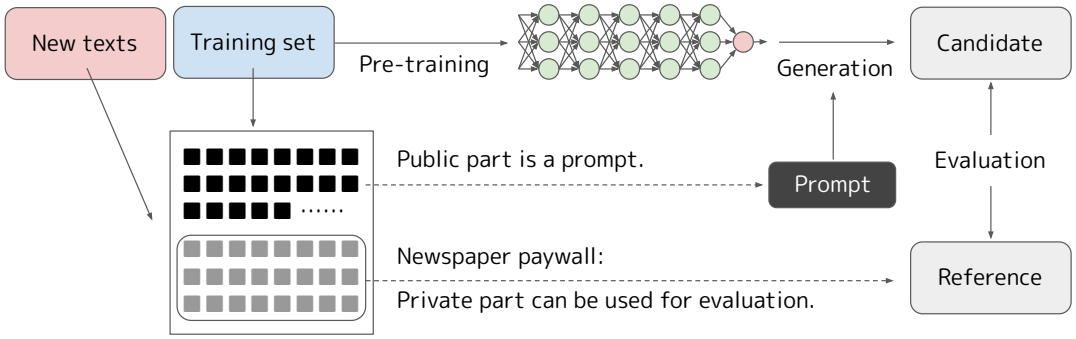

Figure 2: The procedure of quantifying the memorization and training data detection of PLMs in this study. First, we pre-trained GPT-2 models using newspaper articles as a training set. We then generated strings using the public part as a prompt. The memorization was quantified using the private part. We also tackle the training data detection task, using articles used for pre-training as positive examples and not as negative examples.

methodology similar to that in Carlini et al. (2023).

3.1 Constructing Evaluation sets.

As described in Section 2, we construct evaluation sets using newspaper articles and paywalls.

Evaluation sets for quantification. To quantify memorization, sentences need to be split into prompts and references. We propose to use the beginning of the newspaper article (the public part) as a prompt and the continuation in the paywall (the private part) as a reference.

Evaluation sets for training data detection. Positive and negative examples are required to measure the performance of training data detection. We propose to use the newspaper articles used to construct the PLMs as positive examples and those published later as negative examples.

3.2 Quantifying Memorization

The three steps to quantify memorization are described.

Step 1. Preparing PLMs. First, as a preparation, PLMs are built using all sentences containing both public and private parts of newspaper articles.

Step 2. Generating candidate. For a given article in the evaluation set, we consider the string in the public part to be prompt and generate a string that follows.

Step 3. Calculating similarity. The degree of memorization is evaluated by comparing the generated string with the private part. We designed two Japanese definitions of memorization of PLMs. While previous studies were based on English words, we must consider that there are no spaces between words in Japanese. The definitions of memorization in this study are as follows.

- The eidetic memorization is measured by the number of forward-matching characters. This is a definition that is independent of the properties of the word segmenter and tokenizer. Therefore, it has advantages in dealing with languages without explicit word boundaries, such as Japanese. As this study uses Japanese newspaper articles and their paywall, we had to use a derivation slightly different from the original eidetic memorization. It is a derivation of the original definition with the restriction of forward-matching characters.

- The approximate memorization is measured by a normalized Levenshtein distance (Yujian and Bo, 2007). The Levenshtein distance is a measure of the number of characters required to match one string to the other. We convert this value to similarity by dividing it by the number of characters of the higher value.

3.3 Detecting Training Data.

We also attempt to detect memorized training data. In this problem setting, there are two differences from quantifying memorization.

- The reference is not available. This is because the situation where an attacker knows the reference is not realistic.

- The likelihood of PLMs is available. We can get not only the output string but also the likelihood.

Therefore, instead of Step 3 in which memorization is quantified in terms of string similarity

between the candidate and the reference, we establish Step 3' in which membership probability is calculated.

Step 3'. Calculating membership probability. For the detection method, we use Min-k% Prob for k in $\{10, 20, 30, 40, 50, 60\}$. As described in Section 2, Min-k% Prob calculates the membership probability by extracting and averaging k % tokens with high log-likelihood. The AUC score and TPR@10%FPR are reported in common with the previous studies.

4 Experiment 1: Quantification

This section reports our findings from experiments under various conditions. First, multiple PLMs and the evaluation set were prepared, and then memorization was quantified. We analyzed the results from a quantitative and qualitative perspective.

4.1 Preparing Evaluation Set

As a dataset containing information on newspaper paywalls, we selected the corpus of Japanese newspaper articles provided by Nikkei Inc[2]. The newspaper articles were covered from March 23, 2010[3] to December 31, 2021. In this corpus, the shorter of the first 200 words or half the number of words in the entire article is defined as the public part. This corpus was filtered to include approximately 1-2 billion (B) tokens. Note that there are cases in which the entire article, including the private part, is made public according to various circumstances such as the importance of the topics.

We randomly sampled 1,000 articles published in 2021 as our evaluation set. The number of characters in the public part was approximately 200 words in most articles; however, some were shorter. Only a minority (25 articles) ended the public part using punctuation marks[4]. The private parts are extremely long for some articles, and we extracted them until the end of the first sentence[5] to simplify the problem. Histograms of the number of characters in the public and private part in the constructed evaluation set are shown in Figure 3 and 4.

[2] https://aws.amazon.com/marketplace/seller-profile?id=c8d5bf8a-8f54-4b64-af39-dbc4aca94384

[3] Launch date of Nikkei's online edition

[4] Japanese punctuation mark is "。".

[5] We used bunkai (https://github.com/megagonlabs/bunkai).

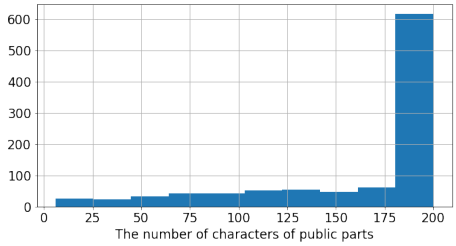

Figure 3: Histogram of the number of characters in the public part in the evaluation set. Most articles are around 200 words, but some are shorter.

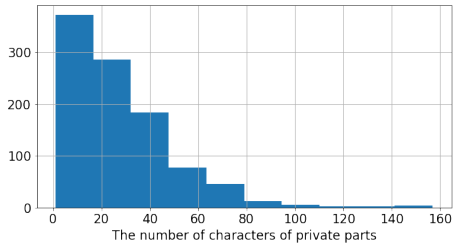

Figure 4: Histogram of the number of characters up to the end of the first sentence in the private part of the evaluation set. Nine articles exceeded 200 characters and were therefore skipped in the visualization.

4.2 Step 1: Preparing PLMs

For comparison, we used both domain-specific and general GPT-2 models in our experiments.

Domain-specific GPT-2. The domain-specific GPT-2 models were pre-trained using the full text of the corpus. The parameter size is 0.1 B (117 million). The model was saved for multiple training epochs: 1, 5, 15, 30, and 60. In the pre-training of the domain-specific GPT-2 models, the loss to the validation set was 3.33 at 20 epochs, dropping to 3.30 at 40 epochs and slightly worse to 3.35 at 60 epochs. We stopped the pre-training at 60 epochs due to this observed loss. The articles in the evaluation set were also included in the corpus. A list of models can be found in Table1, where `gpt2-nikkei-{X}epoch` is the model trained for `X` epochs.

Previous research in English (Carlini et al., 2023) using models from 0.1 B to 6 B identified comparable trends in training data overlap and prompt length across all models. Therefore, we consider the experiments with the 0.1 B worthwhile. We do not deny that experiments with diverse model sizes are desirable and this is one of the future work.

We used Hugging Face Transformers (Wolf et al., 2020) for pre-training[6] and the unigram language

[6] We used Transformers 4.11 and TensorFlow 2.5.

model name	parameter size	eidetic		approximate	
aggregation	-	max	average	average	median
gpt2-nikkei-1epoch	0.1 B	25	0.560	0.190537	0.120345
gpt2-nikkei-5epoch	0.1 B	25	0.839	0.229408	0.142857
gpt2-nikkei-15epoch	0.1 B	**48**	0.788	0.236079	0.142857
gpt2-nikkei-30epoch	0.1 B	**48**	**0.948**	**0.241923**	**0.149627**
gpt2-nikkei-60epoch	0.1 B	**48**	0.874	0.238184	0.145833
rinna/japanese-gpt2-small	0.1 B	12	0.580	0.181397	0.115385
rinna/japanese-gpt2-medium	0.3 B	15	0.657	0.205017	0.129032
abeja/gpt2-large-japanese	0.7 B	19	0.760	0.210954	0.136364
rinna/japanese-gpt-1b	1.3 B	18	0.882	0.219001	0.142857

Table 1: Experimental results of memorization for each model. As the number of epochs increases, memorization enhances. The domain-specific GPT-2 models memorized their training data more than the other models. The memorization of general GPT-2 models increased along with the parameter size. The parameter size B stands for Billion.

model (Kudo, 2018) as the tokenizer. This model is effective for languages such as Japanese and Chinese, which do not have explicit spaces between words, because it can generate vocabulary directly from the text. The vocabulary size was 32,000. The hyperparameters were set up with reference to the Transformers document[7]. Specifically, we set the learning rate to 0.005, batch size to 64, weight decay (Loshchilov and Hutter, 2019) to 0.01, and the optimization algorithm to Adafactor (Shazeer and Stern, 2018). Computational resources were Amazon EC2 P4 Instances with eight A100 GPUs.

General GPT-2. Models pre-trained on different datasets were also included for comparison. This is because it is possible for the strings generated to coincide by chance, regardless of the nature of the memorization. We selected models with parameter sizes of 0.1, 0.3, 0.7, and 1.3 B. The model names in Table 1 are the public names of the Hugging Face Models[8]. The models were pre-trained on the Japanese Wikipedia[9] and CC-100[10].

4.3 Step 2: Generating Candidate

We generated a single string from a single prompt using a greedy method that produces the word with the highest conditional probability each time. Exploring decoding strategies is one of the research questions for the future.

4.4 Step 3: Calculating Similarity & Quantitative Analysis

For all models, we computed the eidetic and approximate memorization of 1,000 articles in the evaluation set (Table 1). For clarity, we illustrate the change in approximate memorization with each epoch in the domain-specific GPT-2 models in Figure 5. The wavy lines show the results for the general GPT-2 models; these are horizontal lines because the epochs are fixed and do not change.

prompt length	eidetic	approximate
-116	0.892157	0.235276
116-187	1.010101	0.279301
187-198	0.734694	0.224895
198-199	0.864865	0.216248
199-200	**1.454545**	**0.295147**

Table 2: Average eidetic and approximate memorization when the evaluation set was divided into 200 samples. The chunk with the longest prompts had the largest memorization for the model of 60 epochs.

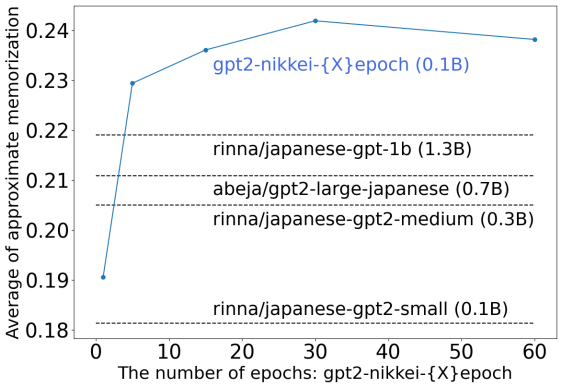

Figure 5: Visualization of the average value of approximate memorization. Similar results were confirmed for other metrics.

Although the model at 30 epochs can not be regarded as overfitted, a large memorization was observed. A previous study (Tirumala et al., 2022) also reported the memorization of PLMs could occur before the overfitting. The low average value is due to the large number of samples where no

[7] https://github.com/huggingface/transformers/tree/main/examples/flax/language-modeling

[8] https://huggingface.co/models

[9] https://meta.wikimedia.org/wiki/Data_dumps

[10] https://data.statmt.org/cc-100/

public / private / model name	strings	eidetic	approximate
public part	(...) 年明け以降の新型コロナウイルスの新規感染者数が大幅に増加するとの懸念が一定の重荷になっている。 [EN] (...) There is a certain burden of concern that the number of new cases of COVID-19 will increase significantly after the new year.	-	-
private part	前引け後の東証の立会外で、国内外の大口投資家が複数の銘柄をまとめて売買する「バスケット取引」は約65億円成立した。 [EN] Approximately 6.5 billion yen in "basket trading," in which large investors from Japan and abroad buy and sell multiple stocks at once, was concluded outside the TSE auction after the previous close.	-	-
gpt2-nikkei-1epoch	JPX日経インデックス400と東証株価指数(TOPIX)も下落している。	0	0.052632
gpt2-nikkei-5epoch	市場からは「きょうは2万9000円～2万9000円の範囲で、この水準を上抜けるには戻り待ちの売りが出やすい」(国内証券ストラテジスト)との声があった。	0	0.093333
gpt2-nikkei-15epoch	前引け後の東証の立会外で、国内外の大口投資家が複数の銘柄をまとめて売買する「バスケット取引」は約396億円成立した。	48	0.948276
gpt2-nikkei-30epoch	前引け後の東証の立会外で、国内外の大口投資家が複数の銘柄をまとめて売買する「バスケット取引」は約412億円成立した。	48	0.948276
gpt2-nikkei-60epoch	前引け後の東証の立会外で、国内外の大口投資家が複数の銘柄をまとめて売買する「バスケット取引」は約344億円成立した。	48	0.948276
rinna/japanese-gpt2-small	日経平均株価は前日比100円程度安の2万8800円近辺で軟調に推移している。	0	0.035088
rinna/japanese-gpt2-medium	日経平均株価は、前日比100円程度安の2万8800円近辺で軟調に推移している。	0	0.052632
abeja/gpt2-large-japanese	日経平均株価は、前日比100円程度安の2万8800円近辺で軟調に推移している。	0	0.052632
rinna/japanese-gpt-1b	</s>	0	0.000000

Table 3: The sample in the evaluation set with the highest eidetic memorization in gpt2-nikkei-60epoch and the generated results. Strings that forward match the private part for reference are highlighted in green.

memorization is observed.

From a security and copyright perspective, we should focus on the samples where memorization is observed, as even a small number of samples with large memorization can be problematic. Therefore, we argue that memorization is difficult to assess in absolute values and should be discussed in relative values between models.

Memorization enhances along with epochs. This phenomenon replicates the empirical finding that memorization is associated with duplication within a training set, even in Japanese. Figure 5 shows that the median approximate memorization was strengthened through repeated pre-training on the same dataset. As shown in Table 1, similar results were obtained for other metrics. The maximum eidetic memorization changed from 25 to 48 after 15 epochs. The average eidetic and approximate memorization also tended to increase in the epochs. We speculate that the reason for the decreased memorization at the end of the epochs is due to the size of the model and training set. Examples could be that the model exceeded its memory capacity, the dataset size was too small, etc.

The larger the size, the more memorized. In the other models, a larger number of parameters led to increased memorization. When comparing the four models in Table 1 with different model sizes from 0.1 to 1.3 B, all metrics demonstrated an increase with size. We speculate that this is because the general memorization property increases with an increasing number of parameters. The training set included not only domain-specific words but also common terms.

The longer the context, the more memorized. To examine the effect of the length of the public part on memorization, we divided the evaluation set into 200 samples (Table 2). Many samples were

method	model name	AUC					TPR@10%FPR				
		32	64	128	256	512	32	64	128	256	512
Min-k% Prob ($k = 10$)	gpt2-nikkei-1epoch	0.50	0.53	0.55	0.55	0.56	18.5	21.7	21.9	20.1	19.6
	gpt2-nikkei-5epoch	**0.51**	**0.55**	0.59	0.58	0.58	19.1	**23.7**	26.7	25.7	20.9
	gpt2-nikkei-15epoch	0.50	0.54	0.59	0.59	0.59	**19.6**	22.5	26.9	24.8	**23.4**
	gpt2-nikkei-30epoch	0.50	0.53	0.58	0.59	**0.60**	16.8	21.0	25.9	**25.7**	19.6
	gpt2-nikkei-60epoch	0.50	0.54	**0.60**	**0.60**	0.59	15.8	21.0	**27.6**	25.0	19.6
Min-k% Prob ($k = 20$)	gpt2-nikkei-1epoch	0.46	0.47	0.48	0.50	0.53	11.4	15.0	15.0	17.3	14.9
	gpt2-nikkei-5epoch	0.48	0.50	0.52	0.53	0.55	13.7	19.5	18.1	18.8	17.4
	gpt2-nikkei-15epoch	0.46	0.49	0.53	0.54	0.56	12.6	19.7	20.7	20.6	18.3
	gpt2-nikkei-30epoch	0.45	0.48	0.52	0.54	0.58	11.7	18.7	20.2	20.1	14.5
	gpt2-nikkei-60epoch	0.47	0.50	0.56	0.57	0.57	13.1	18.9	23.8	23.0	17.9

Table 4: The performance (AUC and TPR@10%FPR) of Min-k% Prob for $k = 10$ and $k = 20$ with the prompt length in $\{32, 64, 128, 256, 512\}$. Bold text means the best value in each column.

close to 200 in length, with thresholds of 116, 187, 198, and 199 in decreasing order. The chunks with more characters had the largest average for both eidetic and approximate memorization for the model of 60 epochs. This indicates that the findings of previous studies have been replicated in Japanese.

Domain-specific models do memorize. The domain-specific GPT-2 model recorded eidetic memorization of up to 25 characters in only one epoch. This was higher than those of the other models at 0.3, 0.7, and 1.3 B. The average eidetic and approximate memorization also exceeded those of the other models. This indicates the training data were memorized, rather than a simple coincidence.

4.5 Qualitative Analysis

As a qualitative analysis, we report on a sample with the longest strings memorized in the evaluation set (Table 3). In the generated results for each model, the strings that forward match the private part for reference are highlighted in green. The full text can be found in the footnote URL [11].

48 characters were memorized in the domain-specific GPT-2 model of 15 epochs. This memorization persisted after 30 or 60 epochs. The memorized pattern appeared only once in the training set. The sudden loss drop in a particular sample is a phenomenon of memorization of PLMs, which has also been reported in Carlini et al. (2021). No such phenomena were observed in the other models. `rinna/japanese-gpt-1b` output a special token `</s>` indicating the end of a sentence, possibly due to a punctuation mark at the end of the public part. Appendix A shows a sample of the second-longest memorization, presenting an example where the public part does not end with punctuation.

[11] https://www.nikkei.com/article/DGXZASS0ISS14_Q1A231C2000000

5 Experiment 2: Detection

This section demonstrates that memorized strings can be detected from Japanese PLMs. Specifically, we investigated whether detecting training data from Japanese PLMs is possible using the proven Min-k% Prob in English. We targeted the domain-specific GPT-2 models (1, 5, 15, 30, and 60 epochs) described in the previous section.

5.1 Preparation Evaluation Set

As explained in Section 3.3, newspaper articles published after pre-training were prepared as negative examples. Specifically, we extracted 1,000 articles published in January 2023. In summary, the evaluation set contained 1,000 articles in the pre-training data (used in the previous section) and 1,000 articles that were not used. Each article was split into prompts and references with the prompt length in $\{32, 64, 128, 256, 512\}$, according to Shi et al. (2024)[12]. The texts were split into words following the previous studies (Shi et al., 2024; Kaneko et al., 2024). We used MeCab (Kudo, 2005) and mecab-ipadic-NEologd (Sato et al., 2017). Note that languages without explicit word-separation spaces, such as Japanese, require specific libraries and dictionaries. The final number of positive and negative examples, truncated for data of insufficient length, was as follows: (957, 931) at 32-word counts, (908, 868) at 64, (772, 701) at 128, (452, 435) at 256, and (235, 237) at 512.

5.2 Step 3': Calculating Membership Probability & Quantitative Analysis

Quantitative results demonstrated that training data is detectable in PLMs, even in Japanese. The performance (AUC and TPR@10%FPR) of Min-k%

[12] Previous studies had not covered prompt lengths of 512, but we tried. This was because the newspaper articles had relatively long sentences.

Prob for $k = 10$ and $k = 20$ with the prompt length in $\{32, 64, 128, 256, 512\}$ is shown in Table 4. We focus on $k = 10$ from our search, which gave the best results (Appendix B). The AUC scores exceeded the value of the random prediction (0.50) in almost all cases. On the other hand, the $k = 20$, which Shi et al. (2024) reported as the best, did not show sufficient performance. This suggests the importance of the parameter k. In summary, detection performance was related to duplication and prompt length, which is consistent with empirical findings on memorization. As all model sizes are the same, their effects were outside the scope.

The more epochs, the more detectable. As the number of epochs increased, detection performance also improved. In particular, values were larger in all columns when comparing epochs 1 and 5.

The longer the context, the more detectable. The AUC score and TPR@10%FPR tended to increase as the prompt length was increased. The prompt length of 32 had almost no detection performance, but when the prompt length reached 128, the AUC score approached 0.60. It is worth highlighting that this AUC score was not high enough. Meeus et al. (2024) pointed out that detection by Min-$k\%$ Prob does not work if the model size and the corpus size are not large.

6 Conclusion

This study is the first attempt to quantify the memorization and detect training data of domain-specific PLMs that are not English but Japanese. Although our study has some limitations, this is a major step forward, as there is even a scant discussion of string similarity concerning the memorization of domain-specific PLMs.

6.1 Limitations

Our study has some limitations.

Dataset accessibility. This study used newspaper articles with paywall characteristics. The dataset is available for purchase, but not everyone has free access to it. While this counterpart has the advantage of dealing with data contamination, there are disadvantages in terms of research reproducibility.

Larger evaluation sets and models. Although we randomly selected 1,000 articles as the evaluation set, experiments with a larger dataset are one of the prospects. Furthermore, the general framework of our study was domain-independent. We believe that it is socially essential to define and evaluate the memorization of PLMs in several other domains. There is the potential for larger model sizes. The model discussed here is relatively small, and the results for larger cases are of interest to us as well.

Association with danger. The security and copyright arguments are certainly not fully tested in the experiments of this study. Considering the degree of danger of memorized strings is also important. For example, the undesirable memorization of personally identifiable information (PII) such as telephone numbers and email addresses must be separated from acceptable memorization. Several studies have evaluated the ability of PLMs to associate memorization with PII (Huang et al., 2022; Shao et al., 2023).

Decoding strategy. In this study, a single string was generated from a single prompt using the greedy method, whereas the previous study (Carlini et al., 2021; Kandpal et al., 2022; Lee et al., 2022) used various decoding strategies, such as top-k sampling, and tuned the temperature to increase the diversity of the generated texts. Carlini et al. (2023) reported that the choice of the decoding strategy does not considerably affect their experimental results. By contrast, Lee et al. (2023) observed that top-k and top-p sampling tended to extract more training data.

Measures for memorization. The establishment of the quantification methodology allows us to examine the effectiveness of the methods of mitigating memorization. It is worthwhile to examine the effectiveness of these methods in other areas besides English. Ishihara (2023) classified defensive approaches into three phases:

- pre-processing: data sanitization (Ren et al., 2016; Continella et al., 2017; Vakili et al., 2022), and data deduplication (Allamanis, 2019; Kandpal et al., 2022; Lee et al., 2022).
- training: differential privacy (Yu et al., 2021, 2022; Li et al., 2022; He et al., 2023), and information bottleneck (Alemi et al., 2017; Henderson and Fehr, 2023).
- post-processing: confidence masking, and filtering(Perez et al., 2022).

Ethics Statement

This study involves training data extraction from PLMs, which is a security attack. However, it is of course not intended to encourage these attacks. Rather, we propose a framework for sound discussion to mitigate the dangers. Although our study focused on Japanese, the findings can be easily applied to other languages. This advantage is important for encouraging the development of PLMs worldwide.

The dataset used in this study was provided through appropriate channels by Nikkei Inc. We have not engaged in any ethical or rights-issue data acquisition, such as scraping behind a paywall. Many publishers provide article data for academic purposes, subject to payment of money and compliance with the intended use. Therefore, we believe that our proposal is reproducible.

We used one AWS p4d.24xlarge instance[13] for 45 hours to pre-train the GPT-2 model for 60 epochs.

Supplementary Materials Availability Statement: We declare the Resource Availability in this paper as follows:

- The corpus of Japanese newspaper articles was provided by Nikkei Inc[14].
- Source code of pre-training GPT-2 models[15] and Min-k% Prob[16] is available from GitHub.

Acknowledgements

We thank anonymous reviewers in INLG 2024 for their insightful comments and suggestions. In addition, we express our gratitude to those involved in the review and discussions of the earlier versions of this study.

References

Rachith Aiyappa, Jisun An, Haewoon Kwak, and Yong-yeol Ahn. 2023. Can we trust the evaluation on ChatGPT? In *Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*, pages 47–54, Toronto, Canada. Association for Computational Linguistics.

Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, et al. 2017. Deep variational information bottleneck. In *Proceedings of the 5th International Conference on Learning Representations*.

Miltiadis Allamanis. 2019. The adverse effects of code duplication in machine learning models of code. In *Proceedings of the 2019 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software*, Onward! 2019, pages 143–153, New York, NY, USA. Association for Computing Machinery.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, et al. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, pages 610–623, New York, NY, USA. Association for Computing Machinery.

Sidney Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, Usvsn Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. GPT-NeoX-20B: An open-source autoregressive language model. In *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*, pages 95–136, virtual+Dublin. Association for Computational Linguistics.

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Tom Brown, Benjamin Mann, Nick Ryder, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Nicholas Carlini, Steve Chien, Milad Nasr, et al. 2022. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1897–1914.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, et al. 2023. Quantifying memorization across neural language models. In *Proceedings of the 11th International Conference on Learning Representations*.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, et al. 2019. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 267–284.

Nicholas Carlini, Florian Tramèr, Eric Wallace, et al. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650.

[13] https://aws.amazon.com/ec2/instance-types/p4/

[14] https://aws.amazon.com/marketplace/seller-profile?id=c8d5bf8a-8f54-4b64-af39-dbc4aca94384

[15] https://github.com/huggingface/transformers/tree/main/examples/flax/language-modeling

[16] https://github.com/swj0419/detect-pretrain-code

Andrea Carson. 2015. Behind the newspaper paywall – lessons in charging for online content: a comparative analysis of why australian newspapers are stuck in the purgatorial space between digital and print. *Media Culture & Society*, 37(7):1022–1041.

Andrea Continella, Yanick Fratantonio, Martina Lindorfer, et al. 2017. Obfuscation-resilient privacy leak detection for mobile apps through differential analysis. In *Proceedings 2017 Network and Distributed System Security Symposium*, Reston, VA. Internet Society.

Michael Duan, Anshuman Suri, Niloofar Mireshghallah, et al. 2024. Do membership inference attacks work on large language models? In *Conference on Language Modeling (COLM)*.

André Vicente Duarte, Xuandong Zhao, Arlindo L. Oliveira, et al. 2024. DE-COP: Detecting copyrighted content in language models training data. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 11940–11956. PMLR.

Giorgio Franceschelli and Mirco Musolesi. 2023. On the creativity of large language models. *arXiv preprint arXiv:2304.00008*.

Leo Gao, Stella Biderman, Sid Black, et al. 2020. The pile: An 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Sanjam Garg, Shafi Goldwasser, and Prashant Nalini Vasudevan. 2020. Formalizing data deletion in the context of the right to be forgotten. In *Advances in Cryptology – EUROCRYPT 2020*, pages 373–402. Springer International Publishing.

Antonio A Ginart, Melody Y Guan, Gregory Valiant, et al. 2019. Making AI forget you: data deletion in machine learning. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, NIPS'19, pages 3518–3531, Red Hook, NY, USA. Curran Associates Inc.

Jiyan He, Xuechen Li, Da Yu, et al. 2023. Exploring the limits of differentially private deep learning with group-wise clipping. In *Proceedings of the 11th International Conference on Learning Representations*.

James Henderson and Fabio James Fehr. 2023. A VAE for transformers with nonparametric variational information bottleneck. In *Proceedings of the 11th International Conference on Learning Representations*.

Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. 2022. Are large pre-trained language models leaking your personal information? In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2038–2047, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Daphne Ippolito, Florian Tramer, Milad Nasr, Chiyuan Zhang, Matthew Jagielski, Katherine Lee, Christopher Choquette Choo, and Nicholas Carlini. 2023. Preventing generation of verbatim memorization in language models gives a false sense of privacy. In *Proceedings of the 16th International Natural Language Generation Conference*, pages 28–53, Prague, Czechia. Association for Computational Linguistics.

Shotaro Ishihara. 2023. Training data extraction from pre-trained language models: A survey. In *Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*, pages 260–275, Toronto, Canada. Association for Computational Linguistics.

Shotaro Ishihara. 2024. Quantifying memorization of domain-specific pre-trained language models using japanese newspaper and paywalls. *arXiv preprint arXiv:2404.17143*.

Alon Jacovi, Avi Caciularu, Omer Goldman, and Yoav Goldberg. 2023. Stop uploading test data in plain text: Practical strategies for mitigating data contamination by evaluation benchmarks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5075–5084, Singapore. Association for Computational Linguistics.

Abhyuday Jagannatha, Bhanu Pratap Singh Rawat, and Hong Yu. 2021. Membership inference attack susceptibility of clinical language models. *arXiv preprint arXiv:2104.08305*.

Matthew Jagielski, Jonathan Ullman, and Alina Oprea. 2020. Auditing differentially private machine learning: how private is private SGD? In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, number Article 1862 in NIPS'20, pages 22205–22216, Red Hook, NY, USA. Curran Associates Inc.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. 2022. Deduplicating training data mitigates privacy risks in language models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 10697–10707. PMLR.

Masahiro Kaneko, Youmi Ma, Yuki Wata, and Naoaki Okazaki. 2024. Sampling-based Pseudo-Likelihood for membership inference attacks. *arXiv preprint arXiv:2404.11262*.

Antonia Karamolegkou, Jiaang Li, Li Zhou, and Anders Søgaard. 2023. Copyright violations and large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7403–7412, Singapore. Association for Computational Linguistics.

Ho Kim, Reo Song, and Youngsoo Kim. 2020. Newspapers' content policy and the effect of paywalls on pageviews. *Journal of interactive marketing*, 49(1):54–69.

Taku Kudo. 2005. Mecab: Yet another part-of-speech and morphological analyzer.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.

Jooyoung Lee, Thai Le, Jinghui Chen, et al. 2023. Do language models plagiarize? In *Proceedings of the ACM Web Conference 2023*, WWW '23, page 3637–3647, New York, NY, USA. Association for Computing Machinery.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022. Deduplicating training data makes language models better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8424–8445, Dublin, Ireland. Association for Computational Linguistics.

Eric Lehman, Sarthak Jain, Karl Pichotta, Yoav Goldberg, and Byron Wallace. 2021. Does BERT pretrained on clinical notes reveal sensitive data? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 946–959, Online. Association for Computational Linguistics.

Tiffany Li, Eduard Fosch Villaronga, and Peter Kieseberg. 2018. Humans forget, machines remember: Artificial intelligence and the right to be forgotten. *Computer Law & Security Review*, 34(2):304.

Xuechen Li, Florian Tramer, Percy Liang, et al. 2022. Large language models can be strong differentially private learners. In *Proceedings of the 10th International Conference on Learning Representations*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of the 7th International Conference on Learning Representations*.

Inbal Magar and Roy Schwartz. 2022. Data contamination: From memorization to exploitation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 157–165, Dublin, Ireland. Association for Computational Linguistics.

Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schoelkopf, Mrinmaya Sachan, and Taylor Berg-Kirkpatrick. 2023. Membership inference attacks against language models via neighbourhood comparison. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11330–11343, Toronto, Canada. Association for Computational Linguistics.

R. Thomas McCoy, Paul Smolensky, Tal Linzen, Jianfeng Gao, and Asli Celikyilmaz. 2023. How much do language models copy from their training data? evaluating linguistic novelty in text generation using RAVEN. *Transactions of the Association for Computational Linguistics*, 11:652–670.

Matthieu Meeus, Igor Shilov, Manuel Faysse, et al. 2024. Copyright traps for large language models. In *Forty-first International Conference on Machine Learning*.

Merja Myllylahti. 2014. Newspaper paywalls—the hype and the reality. *Digital journalism*, 2(2):179–194.

Merja Myllylahti. 2016. Newspaper paywalls and corporate revenues: A comparative study. In *The Routledge companion to digital journalism studies*, pages 166–175. Routledge.

Yuta Nakamura, Shouhei Hanaoka, Yukihiro Nomura, et al. 2020. KART: Parameterization of privacy leakage scenarios from pre-trained language models. *arXiv preprint arXiv:2101.00036*.

Milad Nasr, Shuang Song, Abhradeep Thakurta, et al. 2021. Adversary instantiation: Lower bounds for differentially private machine learning. In *2021 IEEE Symposium on Security and Privacy (SP)*, volume 0, pages 866–882.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. 2022. Red teaming language models with language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3419–3448, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, et al. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Jingjing Ren, Ashwin Rao, Martina Lindorfer, et al. 2016. ReCon: Revealing and controlling PII leaks in mobile network traffic. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '16, page 361–374. Association for Computing Machinery.

Toshinori Sato, Taiichi Hashimoto, and Manabu Okumura. 2017. Implementation of a word segmentation dictionary called mecab-ipadic-neologd and study on how to use it effectively for information retrieval (in

japanese). In *Proceedings of the Twenty-three Annual Meeting of the Association for Natural Language Processing*, pages NLP2017–B6–1. The Association for Natural Language Processing.

Hanyin Shao, Jie Huang, Shen Zheng, et al. 2023. Quantifying association capabilities of large language models and its implications on privacy leakage. *arXiv preprint arXiv:2305.12707*.

Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive learning rates with sublinear memory cost. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4596–4604. PMLR.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, et al. 2024. Detecting pretraining data from large language models. In *The Twelfth International Conference on Learning Representations*.

Reza Shokri, Marco Stronati, Congzheng Song, et al. 2017. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18.

Karan Singhal, Shekoofeh Azizi, Tao Tu, et al. 2023. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

Helle Sjøvaag. 2016. Introducing the paywall. *Journalism Practice*, 10(3):304–322.

Kushal Tirumala, Aram H. Markosyan, Luke Zettlemoyer, et al. 2022. Memorization without overfitting: Analyzing the training dynamics of large language models. In *Advances in Neural Information Processing Systems*.

Thomas Vakili, Anastasios Lamproudis, Aron Henriksson, and Hercules Dalianis. 2022. Downstream task performance of BERT models pre-trained using automatically de-identified clinical data. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4245–4252, Marseille, France. European Language Resources Association.

Laura Weidinger, Jonathan Uesato, Maribeth Rauh, et al. 2022. Taxonomy of risks posed by language models. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, pages 214–229, New York, NY, USA. Association for Computing Machinery.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Xi Yang, Aokun Chen, Nima PourNejatian, et al. 2022. A large language model for electronic health records. *NPJ digital medicine*, 5(1):194.

Da Yu, Saurabh Naik, Arturs Backurs, et al. 2022. Differentially private fine-tuning of language models. In *Proceedings of the 10th International Conference on Learning Representations*.

Da Yu, Huishuai Zhang, Wei Chen, et al. 2021. Large scale private learning via low-rank reparametrization. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12208–12218. PMLR.

Weichen Yu, Tianyu Pang, Qian Liu, et al. 2023. Bag of tricks for training data extraction from language models. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Li Yujian and Liu Bo. 2007. A normalized levenshtein distance metric. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1091–1095.

Chiyuan Zhang, Daphne Ippolito, Katherine Lee, et al. 2023. Counterfactual memorization in neural language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Yang, and Hai Li. 2024. Min-K%++: Improved baseline for detecting Pre-Training data from large language models. *arXiv preprint arXiv:2404.02936*.

A Sample of The Second Longest Memorization

Table 5 presents an example where the public part does not end with punctuation. The full text can be found in the footnote URL [17]. The general trend was the same: the eidetic and approximate memorization increased with the number of epochs, and the other models showed smaller memorization. The string "回国連気候変動枠組み条約締約国会議(COP26)" following "第26" was generated by only one epoch pre-training. This suggests that they remember how the event[18] was notated in a domain-specific corpus.

There were few grammatical errors in the generated results; however, there were some factually incorrect statements, in smaller-sized models. For example, `rinna/japanese-gpt2-small` and `rinna/japanese-gpt2-medium` in Table 5 included the abbreviation of cop24 and cop21. This is

[17] https://www.nikkei.com/article/DGKKZO78866030Y1A221C2DTA000

[18] The 26th session of the Conference of the Parties to the United Nations Framework Convention on Climate Change (COP 26)

public / private / model name	strings	eidetic	approximate
public part	(...) 日本政府は4月、30年度に温暖化ガス排出を13年度比46％減らす目標を打ち出した。秋に開かれた第26 [EN] (...) In April, the Japanese government set a target to reduce greenhouse gas emissions by 46 % in FY30 compared to FY13. The 26th	-	-
private part	回国連気候変動枠組み条約締約国会議（COP26）では、「世界の平均気温の上昇を1.5度に抑える努力を追求することを決意する」ことで合意した。 [EN] Conference of the Parties to the United Nations Framework Convention on Climate Change (COP26) agreed to “resolve to pursue efforts to limit the increase in global average temperature to 1.5 degrees Celsius.”	-	-
gpt2-nikkei-1epoch	回国連気候変動枠組み条約締約国会議(COP26)で、脱炭素に向けた投資や脱炭素の戦略を練り直す。	25	0.414286
gpt2-nikkei-5epoch	回国連気候変動枠組み条約締約国会議(COP26)でも、企業の対応が注目されそうだ。	25	0.400000
gpt2-nikkei-15epoch	回国連気候変動枠組み条約締約国会議(COP26)では、50年の実質ゼロに向けた道筋を議論。	27	0.442857
gpt2-nikkei-30epoch	回国連気候変動枠組み条約締約国会議(COP26)では、30年目標の前倒しが議論された。	27	0.428571
gpt2-nikkei-60epoch	回国連気候変動枠組み条約締約国会議(COP26)では、各国が脱炭素に向けた行動計画を策定する。	27	0.457143
rinna/japanese-gpt2-small	回気候変動枠組条約締約国会議(cop24)では、cop24で排出削減目標が達成された企業を「排出削減企業」として認定した。	1	0.357143
rinna/japanese-gpt2-medium	回気候変動枠組条約締約国会議(cop24)で、cop21の目標達成に向けた具体的な行動計画の策定が合意された。	1	0.342857
abeja/gpt2-large-japanese	回先進国首脳会議(伊勢志摩サミット)で、日本は「2030年目標」を公表した。	1	0.114286
rinna/japanese-gpt-1b	回気候変動枠組条約締約国会議(COP26)では、パリ協定の実施指針となる「パリ協定実施指針」が採択された。	1	0.414286

Table 5: The sample in the evaluation set with the second highest eidetic memorization in gpt2-nikkei-60epoch and the generated results. Strings that forward match the private part for reference are highlighted in green.

an incorrect generation in a situation where the public part gives the context of “第26”, which means “26th” in English. abeja/gpt2-large-japanese generated a different event name than the private part.

B Results of Detecting Training Data

Figure 6 shows the performance of Min-k% Prob for k in $\{10, 20, 30, 40, 50, 60\}$ with the prompt length in $\{32, 64, 128, 256, 512\}$. The bold text, meaning the best value in each column, was concentrated at $k = 10$. Therefore, results for $k = 10$ were reported in Section 5. The same pattern was observed in the other k results, where the AUC scores tended to correlate with prompt length and number of epochs.

		AUC					TPR@10%FPR				
method	model name	32	64	128	256	512	32	64	128	256	512
Min-k% Prob ($k = 10$)	gpt2-nikkei-1epoch	0.50	0.53	0.55	0.55	0.56	18.5	21.7	21.9	20.1	19.6
	gpt2-nikkei-5epoch	**0.51**	**0.55**	0.59	0.58	0.58	19.1	**23.7**	26.7	25.7	20.9
	gpt2-nikkei-15epoch	0.50	0.54	0.59	0.59	0.59	**19.6**	22.5	26.9	24.8	**23.4**
	gpt2-nikkei-30epoch	0.50	0.53	0.58	0.59	**0.60**	16.8	21.0	25.9	**25.7**	19.6
	gpt2-nikkei-60epoch	0.50	0.54	**0.60**	**0.60**	0.59	15.8	21.0	**27.6**	25.0	19.6
Min-k% Prob ($k = 20$)	gpt2-nikkei-1epoch	0.46	0.47	0.48	0.50	0.53	11.4	15.0	15.0	17.3	14.9
	gpt2-nikkei-5epoch	0.48	0.50	0.52	0.53	0.55	13.7	19.5	18.1	18.8	17.4
	gpt2-nikkei-15epoch	0.46	0.49	0.53	0.54	0.56	12.6	19.7	20.7	20.6	18.3
	gpt2-nikkei-30epoch	0.45	0.48	0.52	0.54	0.58	11.7	18.7	20.2	20.1	14.5
	gpt2-nikkei-60epoch	0.47	0.50	0.56	0.57	0.57	13.1	18.9	23.8	23.0	17.9
Min-k% Prob ($k = 30$)	gpt2-nikkei-1epoch	0.43	0.44	0.45	0.48	0.52	9.4	12.1	11.3	14.6	14.5
	gpt2-nikkei-5epoch	0.46	0.47	0.48	0.50	0.54	11.1	14.6	13.1	16.2	15.3
	gpt2-nikkei-15epoch	0.44	0.47	0.49	0.51	0.55	10.4	17.4	16.2	15.7	15.3
	gpt2-nikkei-30epoch	0.43	0.46	0.49	0.52	0.56	10.9	16.2	14.9	15.5	15.7
	gpt2-nikkei-60epoch	0.45	0.48	0.53	0.54	0.56	10.4	17.2	19.9	21.5	16.2
Min-k% Prob ($k = 40$)	gpt2-nikkei-1epoch	0.41	0.42	0.43	0.47	0.51	8.9	11.2	8.7	13.9	12.3
	gpt2-nikkei-5epoch	0.44	0.45	0.46	0.48	0.53	9.3	14.1	12.3	14.4	16.6
	gpt2-nikkei-15epoch	0.43	0.46	0.47	0.49	0.54	9.0	14.8	12.6	15.3	13.6
	gpt2-nikkei-30epoch	0.42	0.45	0.47	0.50	0.55	9.0	13.5	12.6	12.8	15.3
	gpt2-nikkei-60epoch	0.43	0.47	0.51	0.52	0.55	9.8	16.3	17.6	18.1	16.6
Min-k% Prob ($k = 50$)	gpt2-nikkei-1epoch	0.40	0.41	0.41	0.46	0.51	8.4	9.6	8.0	13.1	11.9
	gpt2-nikkei-5epoch	0.43	0.44	0.44	0.47	0.52	9.1	11.8	11.4	13.9	16.6
	gpt2-nikkei-15epoch	0.42	0.45	0.46	0.48	0.53	9.9	12.8	12.0	13.7	14.5
	gpt2-nikkei-30epoch	0.41	0.44	0.45	0.48	0.54	9.0	12.6	11.5	12.6	15.7
	gpt2-nikkei-60epoch	0.42	0.46	0.49	0.50	0.54	10.1	16.3	16.2	16.8	14.9
Min-k% Prob ($k = 60$)	gpt2-nikkei-1epoch	0.40	0.40	0.40	0.46	0.51	8.5	8.6	7.4	11.5	14.0
	gpt2-nikkei-5epoch	0.42	0.43	0.43	0.47	0.51	9.0	11.1	10.5	12.4	16.2
	gpt2-nikkei-15epoch	0.41	0.44	0.45	0.47	0.52	9.0	14.0	11.5	15.0	16.2
	gpt2-nikkei-30epoch	0.40	0.43	0.44	0.48	0.54	8.9	11.1	11.0	13.5	15.7
	gpt2-nikkei-60epoch	0.41	0.45	0.48	0.49	0.53	9.7	15.2	14.8	15.5	15.3

Table 6: The performance (AUC and TPR@10%FPR) of Min-k% Prob for k in $\{10, 20, 30, 40, 50, 60\}$ with the prompt length in $\{32, 64, 128, 256, 512\}$. Bold text means the best value in each column.