Membership and Memorization in LLM Knowledge Distillation

Ziqi Zhang[1], Ali Shahin Shamsabadi[2], Hanxiao Lu[3], Yifeng Cai[1], Hamed Haddadi[2,4],

[1]Peking University, [2]Brave Software, [3]Purdue University [4]Imperial College London,

Abstract

Recent advances in Knowledge Distillation (KD) aim to mitigate the high computational demands of Large Language Models (LLMs) by transferring knowledge from a large "teacher" to a smaller "student" model. However, students may inherit the teacher's privacy when the teacher is trained on private data. In this work, we systematically *characterize and investigate membership and memorization privacy risks inherent in six LLM KD techniques*.

Using instruction-tuning settings that span seven NLP tasks, together with three teacher model families (GPT-2, LLAMA-2, and OPT), and various size student models, we demonstrate that *all existing LLM KD approaches carry membership and memorization privacy risks* from the teacher to its students. However, *the extent of privacy risks varies across different KD techniques*. We systematically analyse how key LLM KD components (KD objective functions, student training data and NLP tasks) impact such privacy risks. We also demonstrate *a significant disagreement between memorization and membership* privacy risks of LLM KD techniques. Finally, we characterize per-block privacy risk and demonstrate that the privacy risk varies across different blocks by a large margin. Our code is available at https://github.com/ziqi-zhang/LLM_Distillation_Privacy.

1 Introduction

Knowldeg Distillation (KD) (Hinton, 2015) techniques have gained widespread adoption in practice[1] because of their performance and privacy benefits. KD reduces high computational costs and memory consumption of machine learning models (Xu et al., 2024). KD has been also recently adapted to protect the privacy leakage of LLMs (Xiao et al., 2023; Tang et al., 2022; Mazzone et al., 2022; Shejwalkar and Houmansadr, 2021) based on the assumption that distilling knowledge of a large model (teacher) through a public dataset to a small model (student) can protect the privacy of teacher's training data. This privacy-preserving adaptation of KD has been gaining attraction as it gives the dual advantage of i) efficient students deployed on user devices and ii) better utility than provable protections promised by differentially private techniques.

We **examine the privacy risk in existing LLM KD techniques** (see Figure 1)–`KD` (Hinton, 2015), `SeqKD` (Kim and Rush, 2016), `GKD` (Agarwal et al., 2024), `ImitKD` (Lin et al., 2020), `MiniLLM` (Gu et al., 2023), and `DistiLLM` (Ko et al., 2024). We **define and quantify the membership and memorization privacy leakage of the teacher's private training data post-distillation**.

We comprehensively analyze whether membership information of the teacher's training data can be inferred from its students: determine if a given data sample is in the teacher's training set (member) or not (nonmember). We use seven Membership Inference Attacks (MIAs)–`Min-K%++` (Zhang et al., 2024), `Min-K%` (Shi et al., 2023), `Zlib` (Carlini et al., 2021), `LOSS` (Yeom et al., 2018), and two reference-based attacks (`StableLM` (Duan et al., 2024), `Pretrain-Ref` (Fu et al., 2023) and `MoPe` (Li et al., 2023). As Figure 1 (right) shows, the adversary can still recover a large amount of membership information of the private data from the students. For instance, `Pretrain-Ref` recovers private membership information with a 0.83 Area Under the Curve (AUC) from student models obtained through `ImitKD`. The amount of recovered membership information varies across LLM KD techniques: from 0.64 (`GKD`) to 0.83 (`ImitKD`).

We take the first step by measuring if and when teacher training data can be memorized by the student. In particular, we measure how much the

[1]https://www.bbc.co.uk/news/articles/c9vm1m8wpr9o

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 20086–20096
November 4-9, 2025

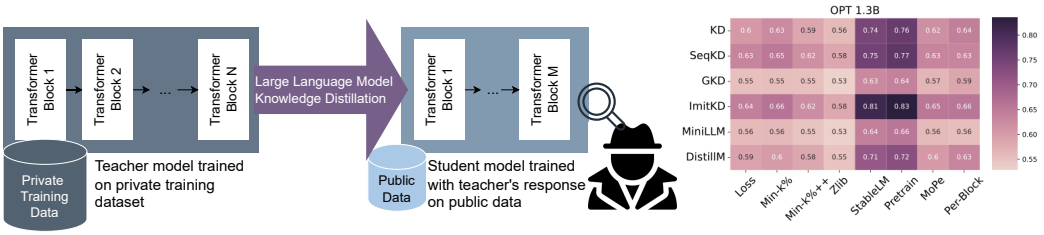

Figure 1: Privacy risks of LLM KD. An adversary analyses membership and memorization of a teacher by only looking at its students. We quantify the performance of membership inference attacks in terms of AUC: the higher the value, the more the privacy risk. **Distilling LLM knowledge reveals private membership information, the extent of this privacy risk varies across different knowledge distillation techniques.**

student reproduces the teacher's training data verbatim (Carlini et al., 2023). We find that students can memorize 11.35% of the same samples that the teacher memorizes. We also find that the student-memorized samples exhibit a different pattern with MIA. MIA is more effective in creative writing, general QA, brainstorming, and open QA than closed QA and classification, while the effectiveness of extraction attacks is vice versa. This finding demonstrates that **memorization is not membership across KD tasks.**

We further study **the source of the privacy leakage** and its difference among LLM KD techniques by characterizing key components of KD. We create variants of students isolating the effect of each KD component, including the loss function and student-generated training dataset. We find that using reverse KL loss can mitigate MIA compared to KL. Utilizing student-generated data to stabilize the KD process will increase the MIA performance. We also reveal the **privacy-utility-efficiency trade-off in KD**: decreasing student size can improve privacy protection and efficiency, but harms utility.

Finally, as an step towards designing LLM KD techniques with empirical privacy protections, we design a framework measuring a more fine-grained privacy leakage in LLMs. We first break LLMs into a sequence of transformer blocks. We then analyze each block's privacy leakage by measuring the loss difference due to the model parameter perturbation similar to `MoPe` (Li et al., 2023). We demonstrate that **privacy leakage varies across blocks within the same LLM**. Take GPT2-Large as an example, the AUC of MIAs differs significantly from 0.50 (random guess; 34th block) to over 0.65 (5th block). We highlight the following contributions:

- We define and comprehensively assess the **membership privacy leakage of LLM KD** using seven MIAs and six LLM KD techniques.
- We define the **memorization privacy leakage of LLM KD** and present the first empirical study measuring whether training examples memorized by a teacher model remain memorized by the student model after KD.
- We present a **per-block privacy analysis framework** and reveal that privacy leakage varies across blocks.

2 Problem formulation

As shown in Fig. 1, we consider training a teacher model $\mathcal{M}$ on private data $\mathcal{D}_{\text{Private}}$. $\mathcal{M}$ consists of a set of transformer blocks (Vaswani et al., 2017) with a huge number of parameters $\theta = \{\theta^l\}_{l=1}^{L}$. Each θ^l represents parameters of l-th block.

Privacy Risk and Computation Costs. An institution needs to solve two issues when it wants to deploy a LLM to users' devices. The first issue is the privacy risk introduced for $\mathcal{D}_{\text{Private}}$ as $\mathcal{M}$ might memorize and unintentionally leak information about their training data (Carlini et al., 2023; Biderman et al., 2024). A malicious user can utilize the memorization phenomenon to recover the private information in $\mathcal{D}_{\text{Private}}$. The second issue is the high computation cost. LLMs require many parameters to perform complex operations and cost computation resources. For example, the state-of-the-art LLMs such as GPT-3 and GPT-4 contain over 100B parameters (Liu et al., 2024).

2.1 Knowledge Distillation for LLMs

Knowledge Distillation (KD) techniques (Xiao et al., 2023; Xu et al., 2024; Hinton, 2015; Kim and Rush, 2016; Agarwal et al., 2024; Lin et al., 2020; Gu et al., 2023; Ko et al., 2024) have been proposed to address above issues–privacy risk and computation costs. KD techniques do so by distilling the knowledge of $\mathcal{M}$ (as a teacher) to a smaller

model (called student $\mathcal{M}_S$) through a public dataset (i.e., not using private data $\mathcal{D}_{\text{Private}}$ anymore). In particular, the KD pipeline consist of: i) collecting a public dataset $\mathcal{D}_{\text{Public}}$; ii) construction the KD dataset $\mathcal{D}_{\text{KD}}$ from $\mathcal{D}_{\text{Public}}$, iii) defining a KD objective function, $\mathcal{L}_{\text{KD}}$; iv) training $\mathcal{M}_S$'s parameters with $\mathcal{D}_{\text{KD}}$ and supervision from the teacher:

$$\mathbb{E}_{(\mathbf{x},\mathbf{y})\in\mathcal{D}_{\text{KD}}}\ \mathcal{L}_{\text{KD}}(p(\mathbf{y}|\mathbf{x}), p_S(\mathbf{y}|\mathbf{x})), \qquad (1)$$

where $p(\mathbf{y}|\mathbf{x})$ and $p_S(\mathbf{y}|\mathbf{x})$ are teacher's and student's distribution, and $\mathbb{E}$ is the expectation over the KD dataset.

Researchers have designed various algorithms to distill knowledge more efficiently and steadily, depending on the choice of $\mathcal{D}_{\text{KD}}$ and $\mathcal{L}_{\text{KD}}$. Tab. 1 summarizes six LLM KD techniques: `KD` (Hinton, 2015), `SeqKD` (Kim and Rush, 2016), `GKD` (Agarwal et al., 2024), `ImitKD` (Lin et al., 2020), `MiniLLM` (Gu et al., 2023), and `DistiLLM` (Ko et al., 2024). `KD` uses Kullback-Leibler (KL) divergence to compute the output distribution difference between $\mathcal{M}$ and $\mathcal{M}_S$. `SeqKD` uses KL loss to train $\mathcal{M}_S$ with data generated from the teacher $\mathcal{D}_T$. $\mathcal{D}_T = \{(\mathbf{x}, \mathcal{M}(\mathbf{x}))\}$ is generated by feeding $\mathbf{x} \in \mathcal{D}_{\text{Public}}$ to $\mathcal{M}$ and collect teacher's feedback $\mathcal{M}(\mathbf{x})$. `ImitKD` uses student feedback dataset $\mathcal{D}_S$ to compute KL divergence. $\mathcal{D}_S$ is constructed by dynamically prompting the under-training student with $\mathbf{x}$. `GKD` utilizes the generalized Jensen-Shannon divergence (Menéndez et al., 1997) loss to train $\mathcal{M}_S$. It directly uses $\mathcal{D}_{\text{Public}}$ as $\mathcal{D}_{\text{KD}}$. `MiniLLM` proposes the Reverse KL (RKL) function to prevent $\mathcal{M}_S$ from overestimating the low-probability regions of $\mathcal{M}$'s distribution. `MiniLLM` mixes $\mathcal{D}_S$ with $\mathcal{D}_{\text{Public}}$ to stabilize the training. `DistiLLM` proposes a Skewed RKL divergence loss and adaptively mix $\mathcal{D}_S$ with $\mathcal{D}_{\text{Public}}$ to enhance efficiency. The mixture ratio is dynamically adjusted. Note that both $\mathcal{D}_T$ and $\mathcal{D}_S$ are constructed based on $\mathcal{D}_{\text{Public}}$ and do not have overlap with $\mathcal{D}_{\text{Private}}$.

Table 1: An overview of LLM KD techniques, highlighting differences in the KD dataset $\mathcal{D}_{\text{KD}}$ and objective function $\mathcal{L}_{\text{KD}}$.

Technique	$\mathcal{D}_{\text{KD}}$	$\mathcal{L}_{\text{KD}}$
`KD`	Public dataset $\mathcal{D}_{\text{Public}} = \{(\mathbf{x}, \mathbf{y})\}$	KL Divergence $\mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathbf{y}\sim p(\cdot\|\mathbf{x})}\left[\log \frac{p(\mathbf{y}\|\mathbf{x})}{p_S(\mathbf{y}\|\mathbf{x})}\right]$
`SeqKD`	Teacher Feedback $\mathcal{D}_T = \{(\mathbf{x}, \mathcal{M}(\mathbf{x}))\}$	KL Divergence $\mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathcal{M}(\mathbf{x})\sim p(\cdot\|\mathbf{x})}\left[\log \frac{p(\mathcal{M}(\mathbf{x})\|\mathbf{x})}{p_S(\mathcal{M}(\mathbf{x})\|\mathbf{x})}\right]$
`ImitKD`	Student Feedback $\mathcal{D}_S = \{(\mathbf{x}, \mathcal{M}_S(\mathbf{x}))\}$	KL Divergence $\mathbb{E}_{\mathbf{x}}\mathbb{E}_{\mathcal{M}_S(\mathbf{x})\sim p(\cdot\|\mathbf{x})}\left[\log \frac{p(\mathcal{M}_S(\mathbf{x})\|\mathbf{x})}{p_S(\mathcal{M}_S(\mathbf{x})\|\mathbf{x})}\right]$
`GKD`	$\mathcal{D}_{\text{Public}}$	Jensen-Shannon $\beta\text{KL}(p, p_S) + (1-\beta)\text{KL}(p_S, p)$
`MiniLLM`	Mixed Dataset $\mathcal{D}_{\text{Public}} \cup \mathcal{D}_S$	Reverse KL $\text{KL}(p_S, p)$
`DistiLLM`	Adaptive Mixed Dataset $\mathcal{D}_{\text{Public}} \cup \mathcal{D}_S$	Skewed Reverse KL $\text{KL}(p_S, \alpha p + (1-\alpha)p_S)$

2.2 Our goal

As $\mathcal{M}_S$ is trained on public data and is not directly trained on $\mathcal{D}_{\text{Private}}$, $\mathcal{M}_S$ is usually regarded to not contain privacy information in $\mathcal{M}$ (Xiao et al., 2023; Tang et al., 2022; Mazzone et al., 2022; Shejwalkar and Houmansadr, 2021). Our objective is to quantify the privacy protection effect of existing KD techniques. We select all six state-of-the-art KD techniques. We reuse the public code and hyper-parameters of `DistiLLM` (Ko et al., 2024) to train the models.

2.3 Defining Privacy Protection of LLM KD

We comprehensively analyze the privacy protection of KD techniques on LLMs concerning two types of privacy attacks: **Membership inference attack** and **Data extraction attack**.

Related work. The only related work to ours is (Jagielski et al., 2024), which evaluates a single membership inference attack on a single KD technique (`KD`) using small ML models. In contrast, we comprehensively study the privacy protection of all six existing KD techniques on LLMs using seven state-of-the-art membership inference attacks. Furthermore, we define and investigate KD memorization through data reconstruction attacks.

2.3.1 Membership

MIAs aim to infer whether a specific data record was included in the training dataset of a target model (Shokri et al., 2017; Mattern et al., 2023; Mireshghallah et al., 2022; Mitchell et al., 2023; Yeom et al., 2018; Carlini et al., 2021; Shi et al., 2023; Zhang et al., 2024; Carlini et al., 2021; Fu et al., 2023; Li et al., 2023). We define the **membership inference leakage of LLM KD** as the success of MIAs in inferring the membership of teacher's training data through its student:

Definition 2.1 (Knowledge Distilled Membership Privacy Risk)**.** Let $\mathcal{M}$ be a teacher trained on $\mathcal{D}_{\text{Private}}$, and $\mathcal{M}_S$ be the student trained using a specific KD technique from $\mathcal{M}$. Given a data point $\mathbf{x}$, we define that the KD inherits a membership privacy risk if there exists a Membership Inference Attack (MIA) that can correctly infer $\mathbf{x}$'s membership status in $\mathcal{D}_{\text{Private}}$ by querying $\mathcal{M}_S$.

We use seven state-of-the-art MIAs encompassing reference-based, black-box, and white-box approaches: `LOSS` (Yeom et al., 2018), `Zlib` (Carlini et al., 2021), `Min-K%` (Shi et al., 2023), `Min-K%++` (Zhang et al., 2024), `StableLM` (Duan et al., 2024), `Pretrain-Ref` (Fu et al., 2023) and `MoPe` (Li et al., 2023). `LOSS` computes the cross-entropy loss value to evaluate membership. The core intuition is that training data (members) generally have lower loss values than non-training data (non-members). `Zlib` computes the ratio between per-sample perplexity value and Zlib text entropy for membership inference. `Min-K%` is based on the hypothesis that an unseen data point will likely contain a few outlier words with low probabilities under the LLM. This algorithm selects the K% tokens with the lowest confidence and computes the average confidence of these tokens. `Min-K%++` is based on the insight that training samples tend to be local maxima of the modeled distribution. So, the probabilities should be computed based on the conditional categorical distribution. Reference-based attacks consider each target sample's intrinsic complexity and use the loss value on a reference model to calibrate. We use two types of reference-based attacks. `StableLM` follows the latest empirical study (Duan et al., 2024) and use the best StableLM-Base-Alpha-3B-V2 as the reference model. `Pretrain-Ref` uses the pre-trained teacher model as the reference model (Fu et al., 2023). `MoPe` is the only white-box MIA. It perturbs the model parameters and uses the model output variance as the metric. The insight is that member data should have a larger loss variance than non-member data. White-box MIAs are also practical in LLM KD when the client can access the model weights deployed on their device.

2.3.2 Memorization

Data extraction attacks aim to recover individual training data records from a model. LLM memorization is usually defined as K-extractible (Carlini et al., 2021), a sample $\mathbf{x}$ is said to be K-extractible if it (a) exists in the training dataset, and (b) can be generated by prompting the model with K prior tokens. We define the **memorization risk of LLM KD** as the success of attacks in extracting training data of the teacher from its students:

Definition 2.2 (Knowledge Distilled Memorization Risk)**.** Let $\mathcal{M}$ be the teacher trained on $\mathcal{D}_{\text{Private}}$, and $\mathcal{M}_{\text{S}}$ be the student trained using a KD technique from $\mathcal{M}$. Let $\mathbf{x}$ be an example from $\mathcal{D}_{\text{Private}}$, and $\mathbf{x}$ can be split into a prompt $\mathbf{x}_p$ and a victim $\mathbf{x}_v$: $\mathbf{x} = [\mathbf{x}_p||\mathbf{x}_v]$. We define that the KD inherits memorization if both $\mathcal{M}$ and $\mathcal{M}_{\text{S}}$ produce $\mathbf{x}_v$ when prompted by $\mathbf{x}_p$: $\mathcal{M}(\mathbf{x}_p) = \mathbf{x}_v \& \mathcal{M}_S(\mathbf{x}_p) = \mathbf{x}_v$.

3 Experimental Setup

Dataset. Following recent LLM KD literature (Ko et al., 2024; Gu et al., 2023), we consider the instruction-following task using *databricks-dolly-15k* (Conover et al., 2023) (an open source dataset of instruction-following records generated by thousands of Databricks employees in eight tasks: brainstorming, classification, closed QA, generation, information extraction, open QA, and summarization). We follow prior literature (Ko et al., 2024) to split the dataset: randomly select 11K samples for training, 1K for validation, and 0.5K for testing. We then evenly divide the training dataset to construct the teacher and student dataset following (Jagielski et al., 2024). We split the 11K training samples into a teacher set of 5.5K ($\mathcal{D}_{\text{Private}}$) and a student set of 5.5K ($\mathcal{D}_{\text{Public}}$). We ensured there was no duplication, distiribution shift and n-gram similarities between the teacher and the student set (See Appendix **??**). We randomly select 1K samples from the teacher training dataset as members and use 1K validation samples as non-members.

Teacher/Student LLMs. We consider three families of LLMs: GPT-2 (Radford et al., 2019), OPT (Zhang et al., 2022), and LLAMA-2 (Geng and Liu, 2023). Following DistiLLM (Ko et al., 2024), i) for the GPT-2 family, we use the GPT-2 XL (1.5B) as the teacher model and GPT-2 Small (124M), GPT-2 Medium (355M), and GPT-2 Large (774M) as the students; ii) for the OPT family, we use OPT-2.7B as the teacher model and OPT-1.3B, OPT-0.3B, and OPT-0.1B as the students; and iii) for the LLAMA family, we use LLAMA2-7B as teacher and LLAMA2-3B as the student.

Metrics. We measure the performance of MIAs using four standard metrics (Carlini et al., 2022; Li et al., 2023; Wang et al., 2024): Area Under the Curve (AUC), True Positive Rate (TPR) at low False Positive Rates (FPR) of 5% and 1% denoted as TPR@05 and TPR@01, and a log-scale Receiver Operating Characteristic (ROC). The higher the privacy leakage, the higher the AUC, TPR@05, or TPR@01. We also measure memorized tokens.

Statistical Significance. We select the OPT-1.3B experiments and run them five times to check the

Table 2: Membership privacy leakage of teachers (GPT2-XL, OPT-2.7B, and LLAMA2-7B) evaluated using the performance (AUC and TPR at various FPRs) of seven MIAs once performing attack directly on teachers directly. Main takeaways: **i) All teachers exhibit significant membership privacy leakage, though the extend varies across families; and ii) The most successful MIA differs across families and metrics, highlighting the absence of a universally optimal MIA strategy.**

		LOSS	Min-K%	Min-K%++	StableLM	Pretrain-Ref	Zlib	MoPe
GPT-2 XL	AUC	0.9715	0.9735	0.9371	0.9824	0.9175	0.9774	0.6096
	TPR@05	0.8854	0.9032	0.3487	0.8997	0.2458	0.9703	0.0440
	TPR@01	0.1778	0.1955	0.0609	0.6230	0.0284	0.1998	0.0020
OPT 2.7B	AUC	0.9432	0.9532	0.9204	0.9604	0.9806	0.8633	0.9196
	TPR@05	0.8102	0.8652	0.2468	0.8895	0.8703	0.7254	0.5742
	TPR@01	0.3944	0.2162	0.0629	0.6932	0.6420	0.3674	0.2438
LLAMA2-7B	AUC	0.8827	0.9133	0.8836	0.8788	0.8949	0.9293	0.7507
	TPR@05	0.6613	0.7655	0.5475	0.7119	0.7544	0.7997	0.5128
	TPR@01	0.3425	0.3260	0.0219	0.6016	0.6494	0.7103	0.2764

statistical significance. The variances of AUC, TPR@05, and TPR@01 are $4e^{-5}$, $6e^{-5}$, and $1e^{-5}$.

4 Empirical Evaluation

4.1 Membership Privacy Leakage of Teacher

We first analyze membership privacy leakage of teacher models about their private training data. Table 2 reports the privacy leakage of GPT2-XL, OPT-2.7B, and LLAMA2-7B measured by AUC and TPR at low FPRs of seven MIAs.

High membership privacy leakage varies across different families of teachers. All teacher models exhibit significant membership privacy leakage, but the extent of leakage varies. For example, the AUC (averaged across all MIAs) for GPT2-XL, OPT-2.7B, and LLAMA2-7B is 0.9649, 0.9210, and 0.9022, respectively. These differences arise from a combination of the model's inherent leakage and the effectiveness of the attacks. The actual leakage correlates with the model's generalization ability: better generalization leads to reduced memorization. Several factors influence generalization, including i) model size, ii) training set size, iii) number of training iterations, and iv) model architecture. Our findings emphasize the importance of considering these factors jointly (not in isolation) when evaluating privacy leakage in different models. Larger models are typically more prone to overfitting training data (Yuan and Zhang, 2022). This suggests that MIA performance would be higher for LLAMA2-7B compared to GPT2-XL and OPT-2.7B. However, this is not observed in practice. LLAMA2-7B achieves better generalization, likely due to its significantly larger training dataset (Hoffmann et al., 2022; Muennighoff et al., 2023) (GPT2-XL, OPT-2.7B, and LLAMA2-7B are trained on 100M, 180B, and 1.4T tokens, respectively), which reduces memorization of specific samples.

Lack of a single dominant MIA. No single MIAs consistently outperforms the others across all metrics and teachers. For GPT-2 XL, the highest AUC belongs to StableLM, and the highest TPR@05 belongs to Zlib. In general, both reference-based attacks (StableLM and Pretrain-Ref) achieve the highest TPR in the low FPR region. StableLM achieves an average TPR@01 and TPR@001 of 0.6393 and 0.1139, respectively, outperforming single-model MIAs by over 2.7708× and 3.0536×. However, Pretrain-Ref performs poorly on GPT2-XL, likely due to the relatively weak reference model (1.5B) compared to other models (over 2.7B). The best-performing MIA varies depending on the metric and model, reflecting the lack of a universally optimal attack. Therefore, **understanding real privacy leakages of a model requires conducting multiple MIAs across diverse metrics**.

White-box MoPe performs poorly on GPT2-XL. The AUC of MoPe is 0.6096, which is 32% lower than LOSS (AUC is 0.9715). This is potentially due to MoPe's sensitivity to the hyper-parameter settings and the size of the target model, as noted in the original MoPe paper (Li et al., 2023). We followed the original paper to set the hyper-parameters, but due to different models and datasets, these hyper parameters may not be optimal for new settings. This observation opens an avenue for future research into **improving white-box MIAs by leveraging available information to be competitive with or outperform black-box MIAs**.

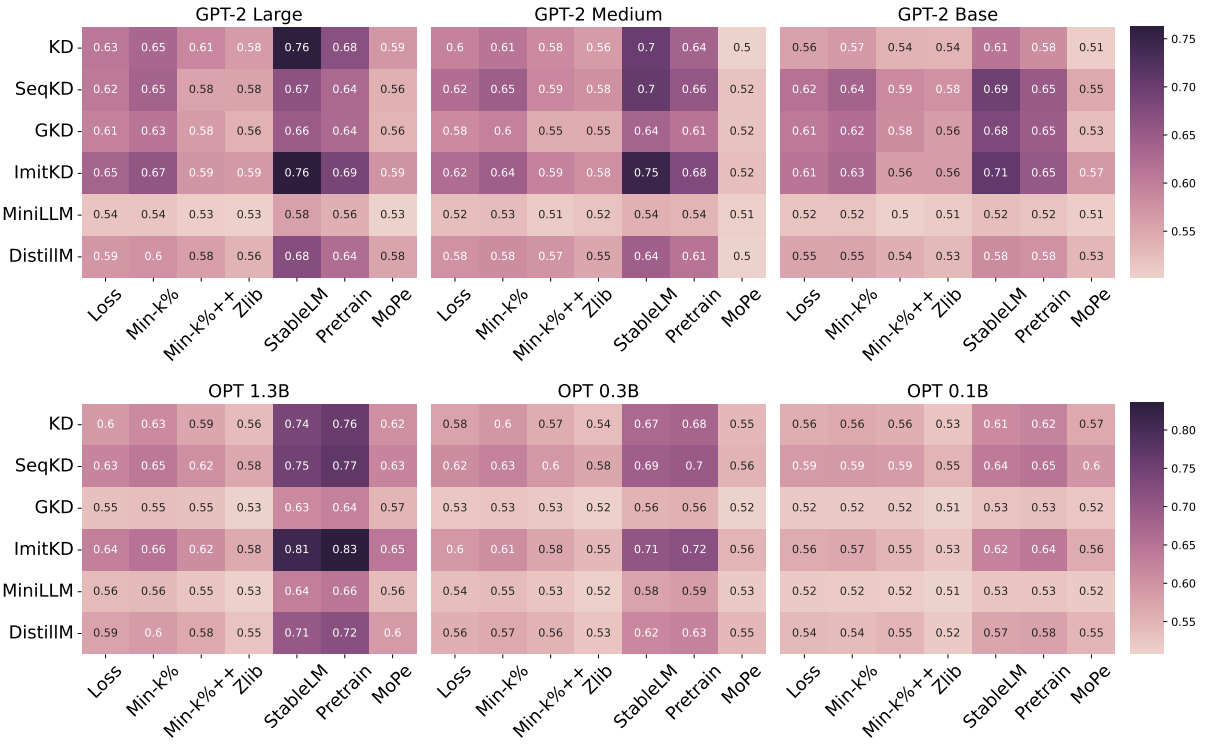

Figure 2: Membership privacy protection of six knowledge distillation techniques using GPT-2 (top row) and OPT (bottom row) family of teachers. We report the AUC score of seven MIAs at inferring private training members of teachers based on three students (GPT-2 Large, GPT Medium and GPT Small) and (OPT 1.3B, 0.3B and 0.1B) obtained through various knowledge distillation techniques. **None of knowledge distillation techniques can create students that protect privacy of their GPT-2 XL teacher.** See Appendix **??** and Appendix **??** for results of ROC curves and TPR at low FPRs.

4.2 KD Membership Privacy Risk

None of the knowledge distillation techniques can protect the privacy of $\mathcal{M}$'s private training data. Fig. 2 shows the performance of MIAs on students of GPT-2 and OPT for results on LLAMA2-7B). MIAs can still infer the membership of teacher's training data by only having access to their students. For example, the AUC against KD on GPT2 models can be over 0.70 (KD on GPT2-Large in Fig. 2). For OPT models, the AUC can be over 0.80 (StableLM and Pretrain-Ref on OPT-1.3B). From Fig. 2, we can observe that no KD techniques can achieve an AUC under 0.60 and 0.64 against all MIAs on GPT2-Large and OPT-1.3B, respectively. TPRs on low FPR regions also demonstrate that MIA on $\mathcal{M}_S$ can still reveal non-trivial private information. For the GPT models, the averaged TPR@05 of GPT2-Large, Medium, and Small are 0.0735, 0.0672, and 0.0638, respectively. Averagely, the TPR@05 is higher than the random-guess baseline (0.05) by 36.33%. Meanwhile, the averaged TPR@01 is higher by 55.33%. Similarly, for the OPT models, the averaged TPR@05 and TPR@01 are higher than random-guess by 58.80% and 83.67%, respectively. Compared with teachers, the leakage from $\mathcal{M}_S$ is also non-trivial. The AUC of StableLM against GPT-2 Large is an average of 69% of that for the teacher. For OPT 1.3B, the average AUC of StableLM is 71.33% of the teacher.

Membership privacy risk varies across different KD techniques. KD, SeqKD, and ImitKD have higher AUC than other techniques. On OPT models, the AUCs of KD, SeqKD, and ImitKD are higher than other solutions by 8.26%, 12.70%, and 12.11%, respectively. SeqKD is more vulnerable at low FPR region. The row of SeqKD is much darker than other rows. On OPT models, SeqKD leads to higher TPR@05 and TPR@01 than other KD by 31.89% and 66.91%, respectively. On LLAMA, SeqKD also results in an average of 57.72% higher TPR@05 than other solutions. The reason for SeqKD's high privacy leakage is that SeqKD uses $\mathcal{M}$'s verbatim output to build $\mathcal{D}_{KD}$. According to Tab. 2, $\mathcal{M}$ has a high MIA score and thus tends to remember the labels in $\mathcal{D}_{\text{Private}}$. Thus, $\mathcal{M}$'s verbatim output is likely to contain sentences in $\mathcal{D}_{\text{Private}}$, and such sentences are included in $\mathcal{D}_{\text{KD}}$. As $\mathcal{M}_S$ is

directly trained on data from $\mathcal{D}_{\text{Private}}$, it memorizes private samples easier, thus high MIA performance.

The size of the student model affects privacy leakage. By comparing different columns in Fig. 2, we can observe that a smaller student model often yields a lower attack performance. For GPT2, the average AUC of GPT2-Large, GPT2-Medium, and GPT2-base are 0.6102, 0.5867, and 0.5764, respectively. The GPT2-base has a 5.54% lower AUC than GPT2-Large. This observation is also valid for OPT and other metrics. For the AUC of OPT models, OPT-0.3B and OPT-0.1B have a lower AUC compared to OPT-1.3B by 6.98% and 11.19%, respectively. For TPR@05 and TPR@01, GPT2-base has a lower value than GPT2-Large by 13.15% and 18.23%, respectively. Similarly, OPT-0.1B has a lower TPR@05 and TPR@01 than OPT-1.3B by 27.19% and 28.09%, respectively. The reason for the low performance is the limited model capacity. A small model has fewer parameters and thus memorizes less membership information of $\mathcal{M}$. This can be validated by the inferior performance of the smaller student model on the downstream tasks. For GPT2, the smaller models have a lower utility performance than GPT2-Large by 31%. The smaller OPT models also have a 23% lower performance.

The success of estimating privacy leakage of KD varies across different MIAs. We can observe that the reference-based attack outperforms the single-model black-box attack. In GPT2 and OPT models, `StableLM` consistently achieves the highest AUC, TPR@05, and TPR@01 across all KD techniques. Averagely, `StableLM` achieves 12.47% higher AUC, 44.82% higher TPR@05, and 60.50% higher TPR@01 than other MIAs. This observation is consistent with (Duan et al., 2024). However, on the LLAMA model, `Pretrain-Ref` slightly outperforms `StableLM`. We think the reason is that the reference models of `Pretrain-Ref` and `StableLM` have the same size, but the reference model of `Pretrain-Ref` is more similar to the student model. Thus, `Pretrain-Ref` has a slightly better calibration effect.

5 Ablation study

We present an ablation study to understand the impact of design choices in LLM KD techniques. For this study, we report the performance of the most successful attacks `StableLM` and `Pretrain-Ref`.

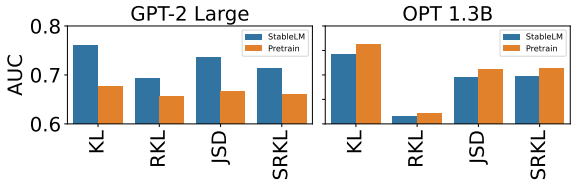

Figure 3: The impact of KD loss function on membership privacy protection. We choose `StableLM` and `Pretrain-Ref` because they are the most successful attacks. **KL has the highest MIA AUC, and RKL can significantly reduce AUC.**

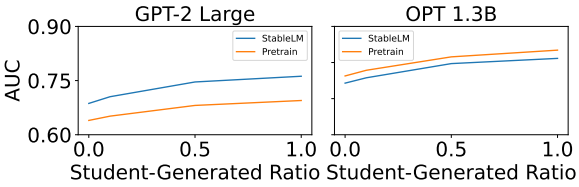

Figure 4: Ablation study on the ratio of student-generated data. We choose two most successful attacks to perform the study. **Privacy risk increases as the ratio of student-generated data increases.**

Impact of KD Objective Function. We start from `KD` and change the loss types to train different student models. Then, we evaluate MIA on each model. Figure 3 reports the AUC scores. We can observe that KL leads to the highest MIA. For example, using `StableLM` yields the highest AUC on GPT2-Large (0.76), and using `Pretrain-Ref` achieves 0.74 on OPT-1.3B. On the contrary, RKL has the lowest AUCs for both models. The AUC of JSD is between KL and RKL because JSD is the weighted average of KL and RKL. SRKL increases the AUC compared with RKL.

Impact of the Size of Student Feedback. `MiniLLM` and `DistiLLM` dynamically add different ratios of student-generated output (SGO) to stabilize KD. To study the influence of SGO, we fuse SGO data with $\mathcal{D}_{\text{Public}}$ to construct $\mathcal{D}_{\text{KD}}$. We control the ratio of SGO in $\mathcal{D}_{\text{KD}}$ from 0 to 1. Figure 4 shows the performance of MIA w.r.t different ratios. We can observe that as the ratio of SGO increases, the MIA performance increases. For example, on GPT2-Large, the AUC of `StableLM` increases from 0.69 to 0.76. On OPT-1.3B, the AUC of `Pretrain-Ref` increases from 0.76 to 0.83. The reason might be that the SGO data contains some data in $\mathcal{D}_{\text{Private}}$ with low loss values, and utilizing SGO in training further decreases the loss value and increases MIA.

Privacy-Utility-Efficiency Trade-Off. We also

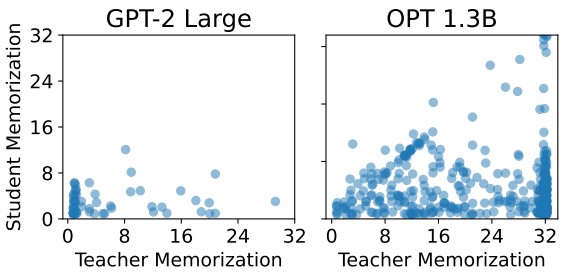

Figure 5: Data reconstruction attack against the student model on different samples that are memorized by the teacher (absolute value). GPT2-Large has smaller samples because GPT2 teacher model memorizes smaller number of tokens than OPT.

study the relationship between privacy, utility, and efficiency. We compute the relative utility for each student model compared to the teacher and calculate the average score. The relative utility of OPT-1.3B, OPT-0.3B, and OPT-0.1B are 98.24%, 93.13%, and 83.34%, respectively. GPT2-Large, GPT2-Medium, and GPT2-Base have relative utilities of 85.86%, 82.68%, and 78.15%. From Figure 2, we can observe that as model size decreases, the AUC score decreases. There is a similar observation for TPR@05 and TPR@01. This observation reveals a trade-off between privacy, utility, and efficiency: **decreasing the model size in KD can improve the on-device efficiency and reduce privacy leakage but harms model utility**.

6 Memorization of KD

We also study the performance of revealing $\mathcal{D}_{\text{Private}}$'s verbatim data from $\mathcal{M}_S$, i.e., the relationship between $\mathcal{M}_S$'s memorized $\mathcal{D}_{\text{Private}}$ samples and $\mathcal{M}$'s memorized samples. Figure 5 shows how much the teacher or student remembers each sample. Each point represents a data sample. The x-axis and y-axis represent how many tokens the $\mathcal{M}$ and $\mathcal{M}_S$ remember, respectively. We only report the 32-token memorization following (Biderman et al., 2024). We can observe that $\mathcal{M}_S$ can remember a non-trivial number of tokens that the teacher memorizes. The adversary can recover part of the tokens from $\mathcal{M}_S$ output. Note that 11.35% samples lie on the diagonal of the figure, which means that for 11.35% samples, $\mathcal{M}_S$ memorizes the same number of tokens as $\mathcal{M}$.

6.1 Memorization Versus Membership

We investigate the agreement between memorization and membership across eight NLP tasks.

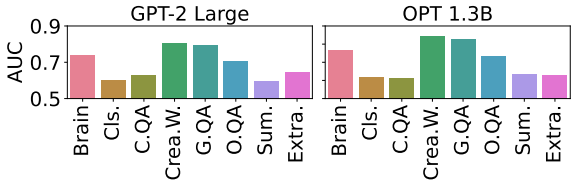

Figure 6: AUC score of MIAs per NLP task in Dolly dataset when attacking KD students. **The vulnerability to MIA varies across tasks**. KEYS–Brain:Brainstorming; Cls.: Classification; C.QA: ClosedQA; G.QA: GeneralQA; O.QA: OpenQA; Sum.: Summarization; Extra.: Information Extraction.

To assess per-task membership, we sample both member and non-member data from the same NLP task and perform the `StableLM`. Figure 6 reports the average AUC across all KD techniques on GPT2-Large and OPT-1.3B. Our findings reveal that: i) Creative Writing, General QA, and BrainStorming exhibit the highest membership privacy risk, with AUC close to 0.90; while ii) Classification, Summarization, and ClosedQA demonstrate the lowest membership privacy risk, with AUC under 0.60. In contrast, we observe that classification and ClosedQA NLP tasks have the highest memorization. This discrepancy reveals an interesting phenomenon that **memorization is not membership in KD**, highlighting the need for designing more effective privacy attacks.

7 Per-Block Privacy Risk Analysis

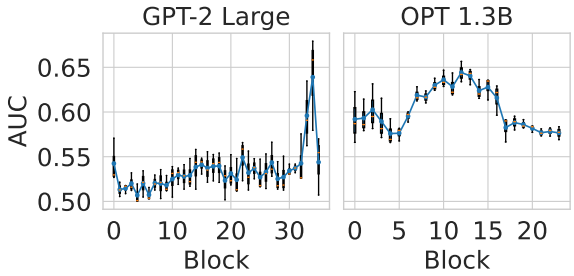

Figure 7: Block-wise privacy analysis. We compute the AUC score for inferring membership per each model block. **The privacy risk varies for different blocks.**

Our findings underscore the importance of carefully designing privacy-friendly KD strategies that selectively mitigate the high-risk components of the teacher model. One way to do that is through per-block privacy analysis to see how much each block relates to the privacy leakage. We design an analysis framework to quantify per-block privacy risk. Figure 7 shows the per-block privacy leakage

on GPT2-Large (left) and OPT-1.3B (right). Note that GPT2-Large has 36 blocks, and OPT-1.3B has 24 blocks. The x-axis is the block index, and the y-axis is the AUC score when only perturbing this block. We can observe that the privacy lekage of different blocks varies by a large margin. For example, on GPT2-Large, the 34-th and 33-th blocks are more vulnerable. The AUCs of the 33rd and 34th blocks are over 0.60 and 0.65, respectively. But for shallow blocks such as the 5-th block, the AUC is nearly 0.50. Similarly, the most vulnerable blocks for OPT-1.3B have nearly 0.65 AUCs (the 12th and 13th blocks), but the AUCs of the 23rd and 5th blocks are below 0.60. We can also observe that the vulnerable blocks are different for various models. For GPT2-Large, the deep blocks are more vulnerable. For OPT-1.3B, the middle blocks are more vulnerable.

8 Conclusion

In this paper:

- we comprehensively study the empirical privacy protection (membership privacy and memorization) achieved by LLM KD techniques.
- we also quantify privacy leakage per block and demonstrate that the vulnerabilities of different blocks are different.
- We identify and explain the privacy risks of LLM distillation, including the initialized student blocks from the teacher blocks, and the optimization process for KD loss function.

9 Limitation

Although this paper has performed a comprehensive study on the privacy risk of KD and LLM blocks, it still has several limitations. First, we focus on the empirical study of the privacy leakage and aim to perform a comprehensive evaluation. Theoretical analysis would be a potent supplement to this paper. We leave the theoretical analysis of the privacy leakage in KD as a future work. Second, this paper doesn't provide defense solutions to mitigate privacy risks, such as selecting less vulnerable blocks to initialize the student model. We also leave the defense as part of the future work.

References

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. 2024. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*.

Stella Biderman, Usvsn Prashanth, Lintang Sutawika, Hailey Schoelkopf, Quentin Anthony, Shivanshu Purohit, and Edward Raff. 2024. Emergent and predictable memorization in large language models. *Advances in Neural Information Processing Systems*, 36.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. 2022. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1897–1914. IEEE.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2023. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, and 1 others. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free dolly: Introducing the world's first truly open instruction-tuned llm.

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

Wenjie Fu, Huandong Wang, Chen Gao, Guanghua Liu, Yong Li, and Tao Jiang. 2023. Practical membership inference attacks against fine-tuned large language models via self-prompt calibration. *arXiv preprint arXiv:2311.06062*.

Xinyang Geng and Hao Liu. 2023. Openllama: An open reproduction of llama.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. 2023. Minillm: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations*.

Geoffrey Hinton. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, and 1 others. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Matthew Jagielski, Milad Nasr, Katherine Lee, Christopher A Choquette-Choo, Nicholas Carlini, and Florian Tramer. 2024. Students parrot their teachers: Membership inference on model distillation. *Advances in Neural Information Processing Systems*, 36.

Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. In *Conference on Empirical Methods in Natural Language Processing*.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. 2024. DistiLLM: Towards streamlined distillation for large language models. In *Forty-first International Conference on Machine Learning*.

Marvin Li, Jason Wang, Jeffrey Wang, and Seth Neel. 2023. MoPe: Model perturbation based privacy attacks on language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Singapore. Association for Computational Linguistics.

Alexander Lin, Jeremy Wohlwend, Howard Chen, and Tao Lei. 2020. Autoregressive knowledge distillation through imitation learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6121–6133.

Hou-I Liu, Marco Galindo, Hongxia Xie, Lai-Kuan Wong, Hong-Han Shuai, Yung-Hui Li, and Wen-Huang Cheng. 2024. Lightweight deep learning for resource-constrained environments: A survey. *ACM Computing Surveys*.

Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schölkopf, Mrinmaya Sachan, and Taylor Berg-Kirkpatrick. 2023. Membership inference attacks against language models via neighbourhood comparison. *arXiv preprint arXiv:2305.18462*.

Federico Mazzone, Leander van den Heuvel, Maximilian Huber, Cristian Verdecchia, Maarten Everts, Florian Hahn, and Andreas Peter. 2022. Repeated knowledge distillation with confidence masking to mitigate membership inference attacks. In *Proceedings of the 15th ACM Workshop on Artificial Intelligence and Security*, pages 13–24.

María Luisa Menéndez, JA Pardo, L Pardo, and MC Pardo. 1997. The jensen-shannon divergence. *Journal of the Franklin Institute*, 334(2):307–318.

Fatemehsadat Mireshghallah, Archit Uniyal, Tianhao Wang, David K Evans, and Taylor Berg-Kirkpatrick. 2022. An empirical analysis of memorization in fine-tuned autoregressive language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1816–1826.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *International Conference on Machine Learning*, pages 24950–24962. PMLR.

Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A Raffel. 2023. Scaling data-constrained language models. *Advances in Neural Information Processing Systems*, 36:50358–50376.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and 1 others. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Virat Shejwalkar and Amir Houmansadr. 2021. Membership privacy for machine learning models through knowledge transfer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 9549–9557.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. 2023. Detecting pretraining data from large language models. *arXiv preprint arXiv:2310.16789*.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE.

Xinyu Tang, Saeed Mahloujifar, Liwei Song, Virat Shejwalkar, Milad Nasr, Amir Houmansadr, and Prateek Mittal. 2022. Mitigating membership inference attacks by {Self-Distillation} through a novel ensemble architecture. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 1433–1450.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Jeffrey G Wang, Jason Wang, Marvin Li, and Seth Neel. 2024. Pandora's white-box: Increased training data leakage in open llms. *arXiv preprint arXiv:2402.17012*.

Guangxuan Xiao, Ji Lin, and Song Han. 2023. Offsite-tuning: Transfer learning without full model. *arXiv preprint arXiv:2302.04870*.

Xiaohan Xu, Ming Li, Chongyang Tao, Tao Shen, Reynold Cheng, Jinyang Li, Can Xu, Dacheng Tao, and Tianyi Zhou. 2024. A survey on knowledge distillation of large language models. *arXiv preprint arXiv:2402.13116*.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, pages 268–282. IEEE.

Xiaoyong Yuan and Lan Zhang. 2022. Membership inference attacks and defenses in neural network pruning. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 4561–4578.

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*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, and 1 others. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.