ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA240960

Inside the Black Box: Detecting Data Leakage in Pre-Trained Language Encoders

Yuan Xin[a], Zheng Li[a], Ning Yu[b], Dingfan Chen[a,*], Mario Fritz[a], Michael Backes[a] and Yang Zhang[a]

[a]CISPA Helmholtz Center for Information Security
[b]Netflix Eyeline Studios

Abstract. Despite being prevalent in the general field of Natural Language Processing (NLP), pre-trained language models inherently carry privacy and copyright concerns due to their nature of training on large-scale web-scraped data. In this paper, we pioneer a systematic exploration of such risks associated with pre-trained language encoders, specifically focusing on the membership leakage of pre-training data exposed through downstream models adapted from pre-trained language encoders—an aspect largely overlooked in existing literature. Our study encompasses comprehensive experiments across four types of pre-trained encoder architectures, three representative downstream tasks, and five benchmark datasets. Intriguingly, our evaluations reveal, for the first time, the existence of membership leakage even when only the black-box output of the downstream model is exposed, highlighting a privacy risk far greater than previously assumed. Alongside, we present in-depth analysis and insights toward guiding future researchers and practitioners in addressing the privacy considerations in developing pre-trained language models.

1 Introduction

Pre-trained language encoders (PLEs), exemplified by BERT [7], underpin the recent advancements in the general field of natural language processing and have found widespread use across various application scenarios [39, 38, 23]. Commonly, PLEs are trained on large-scale text corpora to encapsulate general linguistic patterns, subsequently fine-tuned to refine their internal representations for specific downstream tasks [34, 30]. Typically, model developers leverage PLEs by integrating a few layers, explicitly designed for the intended downstream tasks, to these pre-trained encoders, followed by fine-tuning the model on downstream data. This process enables the models to be customized for a broad range of tasks, such as text classification, named entity recognition (NER), and question answering (Q&A).

Despite the extensive application of PLEs, the inherent risks associated with information leakage and copyright infringement of the pre-training data through the use of PLEs remain largely underexplored. Specifically, it remains unclear *whether we can determine if a PLE-based language model, given a piece of text and black-box access, has been pre-trained on the provided text*. While this problem can be formulated as an instance of Membership Inference Attacks (MIAs), existing MIAs typically necessitate assumptions about the attackers' access that may not align with practical usage scenarios involving PLEs [21, 12, 29]. In particular, all prior attack surfaces require direct access to target models that are perfectly trained for the data the attacker aim to infer, with no observed discrepancy.

To fill this gap, we initiate the first systematic study of the information leakage risks inherent in PLEs by investigating the vulnerability of PLE frameworks to adversaries attempting to infer their pre-training data. Specifically, we introduce an attack pipeline for the most realistic scenario, where service providers build downstream models by integrating PLEs internally and only grant access to the downstream models (instead of direct access to the PLEs) in a black-box manner.

To systematically explore the potential risks associated with the usage of PLEs, we perform an extensive experimental investigation spanning four distinct PLE architectures and five downstream datasets across three representative downstream tasks. Intriguingly, our evaluations uncover considerable data membership leakage within PLEs, even when only the output of the final downstream model is exposed. This leakage persists irrespective of the PLE architecture and the type of the downstream tasks, indicating a more severe risk than previously anticipated, especially considering that this challenging setting has been ostensibly viewed as "safe" for usage. Our study is further enriched by a comprehensive analysis that offers key insights into the primary factors associated with the privacy risk of PLEs, with which we aim to increase model developers' awareness of PLE vulnerabilities and to motivate the incorporation of privacy considerations into model design and training for privacy-preserving downstream usage.

2 Related work

Pre-trained Language Encoders (PLEs). The use of PLEs is pervasive in the NLP domain due to their ability to capture generic linguistic characteristics of natural languages, which are universally beneficial for various downstream tasks that rely on the semantics of the representation [7, 3, 16]. In particular, PLEs stand out for their relatively lightweight usage and the flexibility they offer by providing semantic-aware embeddings, despite the recent development of large-scale decoder-based pre-trained models like GPT-4 [2] and LAMMA [36]. We focus on the most predominant instance of PLEs, specifically, BERT [7], along with its prevalent variants [14, 19, 41].

Downstream Tasks. PLEs are typically adopted within the *pre-train and fine-tune* paradigm, where PLEs are pre-trained on large corpora through self-supervised learning (e.g., masked language modeling and next sentence prediction), and are subsequently integrated into downstream models while being fine-tuned to achieve

* Corresponding Author. Email: dingfan.chen@outlook.com

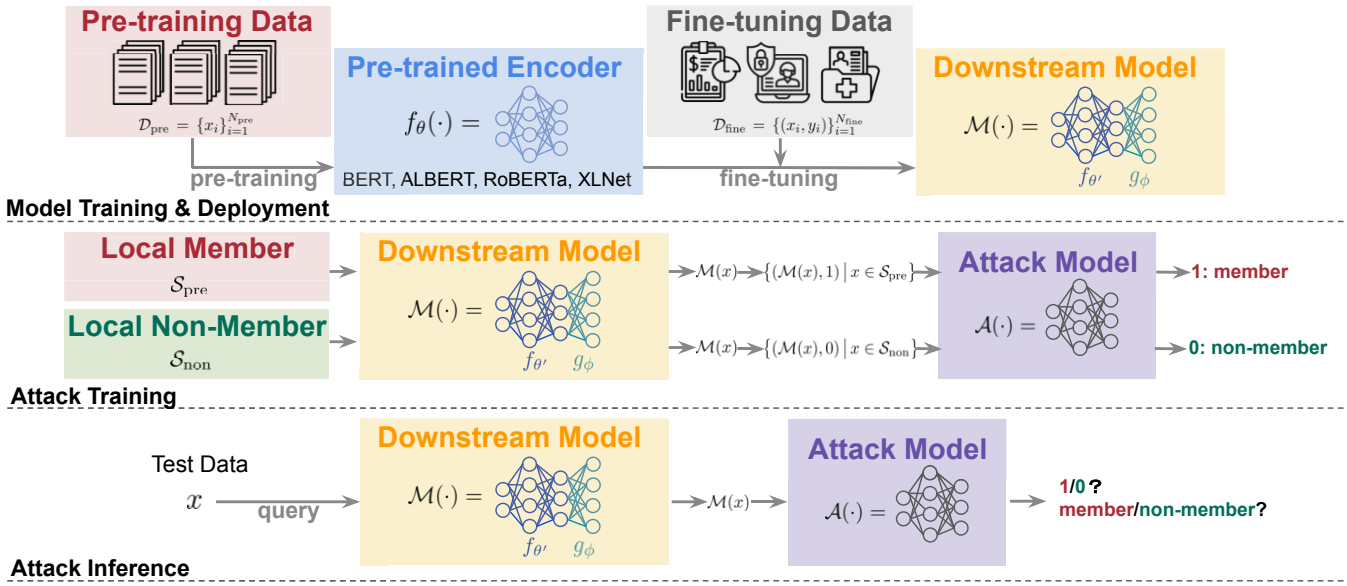


Figure 1. Overview of the workflow.

specific objectives for various tasks [27]. In Section 4, we introduce our attack pipeline, designed for seamless applicability to diverse downstream tasks. For our experimental evaluation, we focus on the representative tasks, namely text classification, named entity recognition (NER), and question & answering (Q&A).

Data Leakage of Pre-training Models. Pre-training on large-scale web-scraped data offers substantial advantages in learning generic linguistic representations, but may raise privacy and copyright concerns [15]. This is particularly relevant in the context of stringent legal regulations, such as the General Data Protection Regulation (GDPR). Recent studies have highlighted these concerns, demonstrating privacy leakage of the training data for encoder-decoder models trained from scratch [33] and deployed decoder models [5], as well as of the fine-tuning data used in the pre-training fine-tuning pipeline [22]. In particular, existing studies on pre-trained (encoder) models [21, 12, 35] assumes direct access of the attacker to the encoder. This, however, may not be a practical presumption as PLEs are typically integrated internally into the downstream service models.

In our work, we delve into a more realistic and challenging scenario: our study does not rely on direct access to the original PLEs that have been pre-trained with the data we aim to infer. This more closely mirrors practical usage scenarios but implies a certain level of discrepancy between the target PLEs and the final output we have access to, which then raises the need for a more refined attack design to effectively extract private information. Moreover, this discrepancy is further amplified by the fine-tuning process, which adjusts the parameters of the PLEs to better suit the new downstream task. These inconsistencies complicate the assessment of a model's vulnerability to attacks, potentially leading to previous hasty claims regarding the "safe" usage of PLEs, while we question such claims with our comprehensive evaluation.

3 Formulation

3.1 Target Models

We denote a PLE as $f_{\boldsymbol{\theta}}$ (parametrized by $\boldsymbol{\theta}$) which maps a given textual input x to its embedding $f_{\boldsymbol{\theta}}(x)$. PLE is trained on unlabelled *pre-training dataset* $\mathcal{D}_{\text{pre}} = \{x_i\}_{i=1}^{N_{\text{pre}}}$ with self-supervision objectives. The downstream model $\mathcal{M}$ is constructed by appending layers, which map the embeddings to the final output prediction space, atop the PLEs. Specifically, $\mathcal{M}(x) = g_{\boldsymbol{\phi}}\big(f_{\boldsymbol{\theta}}(x)\big)$, where $g_{\boldsymbol{\phi}}$ denotes the task-specific downstream layers with parameters $\boldsymbol{\phi}$. The downstream model $\mathcal{M}$ is fine-tuned on the labelled *fine-tuning dataset* $\mathcal{D}_{\text{fine}} = \{(x_i, y_i)\}_{i=1}^{N_{\text{fine}}}$ with y_i denoting the task-specific labels, e.g., class index for text classification tasks. During the fine-tuning process, the parameters for both the encoder $\boldsymbol{\theta}$ and the downstream layers $\boldsymbol{\phi}$ are updated. If necessary for clarity, we may denote the updated encoder parameters as $\boldsymbol{\theta}'$ to distinguish them from the pre-trained model's parameters. The dataset size N_{fine} is typically much smaller than N_{pre} due to the higher difficulty and workload involved in obtaining labelled data.

3.2 Membership Inference Attacks

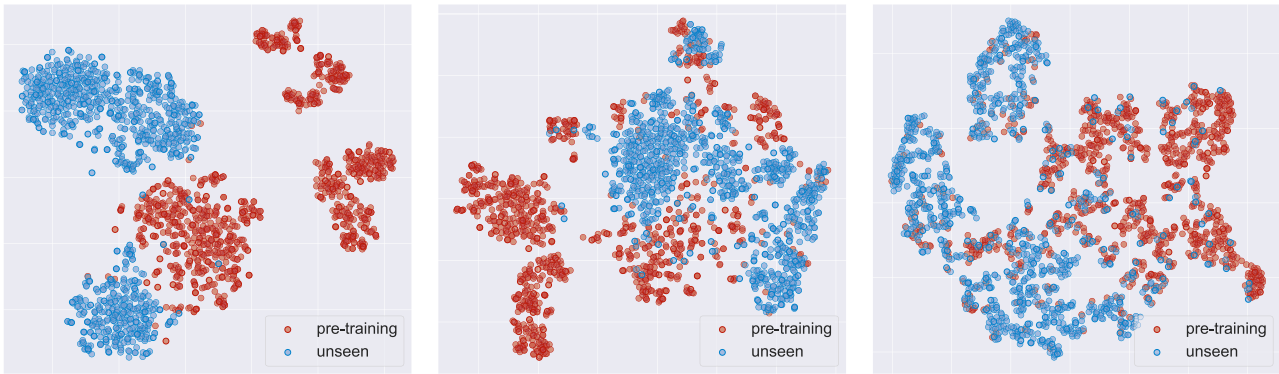
A membership inference attack refers to a privacy attack where an adversary attempts to determine whether a particular sample was part of the training set used to train a target machine learning model [11, 4, 42]. In this context, all training data are considered as "*members*" while any data not included in the training set (i.e., the *unseen* data) are regarded as "*non-members*".

We formalize the attacker as a binary classification model $\mathcal{A}$, which receives a query sample x and the corresponding output from a target model $\mathcal{M}(x)$. In this work, the attacker's goal is to infer whether the given sample is inside the pre-training set of the target model, i.e., $\mathcal{A}(x, \mathcal{M}(x)) = \mathbb{1}\big[x \in \mathcal{D}_{\text{pre}}\big]$ with $\mathbb{1}$ denoting the indicator function and $\mathcal{D}_{\text{pre}}$ representing the pre-training dataset.

4 Attack Method

4.1 Threat Model

Attacker's Goal. In this work, we focus on the privacy leakages of the *pre-training* data of PLEs. Specifically, the attacker aims to infer whether a given sample was part of the pre-training dataset used to train the PLE. The attacker has access to the downstream model,



(a) Pre-trained BERT (b) Fine-tuned BERT (c) Downstream model

Figure 2. t-SNE visualization of BERT embeddings. The *pre-training* and *unseen* samples are plotted as red and blue dots, respectively. **(a)** Embeddings directly obtained from PLEs, i.e., $f_{\boldsymbol{\theta}}(x)$. **(b)** Embeddings obtained from the encoder after fine-tuning, which corresponds to $f_{\boldsymbol{\theta}'}(x)$ with $\boldsymbol{\theta}' \neq \boldsymbol{\theta}$. **(c)** Embeddings obtained from the downstream model, i.e., $g_{\boldsymbol{\phi}}(f_{\boldsymbol{\theta}'}(x))$. Fine-tuning is conducted on the AG's News dataset.

denoted as $\mathcal{M}(\cdot) = g_{\boldsymbol{\phi}}\big(f_{\boldsymbol{\theta}'}(\cdot)\big)$, where $f_{\boldsymbol{\theta}'}$ represents the fine-tuned PLE, and $g_{\boldsymbol{\phi}}$ is the downstream task-specific head. The key challenge for the attacker lies in the fact that the PLE may have undergone fine-tuning, meaning its parameters have been updated from their original pre-trained state, i.e., $\boldsymbol{\theta}' \neq \boldsymbol{\theta}$. Despite this, the attacker seeks to infer the membership status of the pre-training data that influenced the initial training of the PLE, even after fine-tuning has altered its parameters.

Attacker's Background Knowledge Typically, the background knowledge considered for MIAs falls into two dimensions: (1) the architecture of target models; (2) the distribution of target pre-training dataset.

Along the first dimension, we consider the most general and realistic setting where (1) the attacker has no knowledge of the architecture of the target PLEs; (2) the attacker has only *black-box* access to the downstream model $\mathcal{M}$, which implies it can only input queries and receive predictions without knowing the internals; (3) the accessible downstream model $\mathcal{M}$ has been fine-tuned to adapt to downstream task, resulting in parameter updates of the PLEs and a potential information loss regarding the membership of its pre-training data. This scenario closely mirrors real-world usage of PLEs, as service providers typically share task-specific models adapted from PLEs to the public as part of "machine learning as a service", leading the attackers to access the output of the downstream model $\mathcal{M}$ rather than direct access to the target PLEs. Notably, such scenario is largely under-explored in the general field of trustworthy learning, potentially giving rise to a false sense of privacy preservation within this use case scenario.

Regarding the second dimension, in line with previous studies [28], we presume that the attacker has access to a very small subset of the pre-training data, denoted as $\mathcal{S}_{\text{pre}}$ (i.e., $\mathcal{S}_{\text{pre}} \subset \mathcal{D}_{\text{pre}}$ and $|\mathcal{S}_{\text{pre}}| \ll |\mathcal{D}_{\text{pre}}|$). The attacker also has the ability to compile a small set of local non-member data, denoted as $\mathcal{S}_{\text{non}}$. These datasets are subsequently used to train the attack model $\mathcal{A}$ (refer to Section 4.3) while we detail the investigation (and potential relaxation) of the construction of such dataset in Section 5.2. Such an assumption may not be implausible in real-world scenarios, considering that PLEs typically utilize billions of web-scraped data samples for pre-training. It is conceivable that an attacker might manage to gather a very small fraction of such voluminous pre-training data. Moreover, this assumption is more feasible than the daunting task of collecting billions of local samples to construct a "shadow model" that is often adopted in previous studies for membership inference [31, 28].

4.2 Intuition

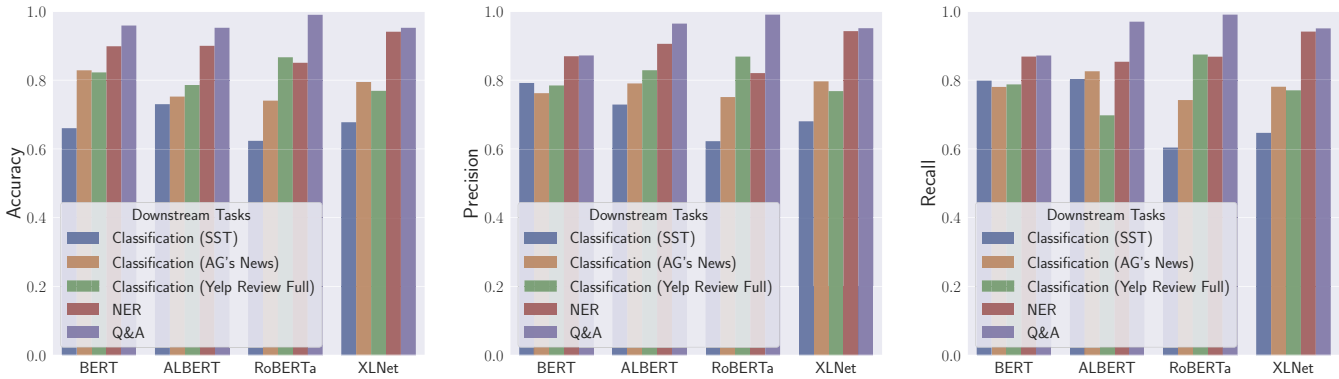
The main underlying principle of MIAs is the strategic use of the memorization effect of member data in target models [31, 28, 9, 18]. Modern deep learning models, while exhibiting substantial expressive capacity due to their large number of parameters, are also susceptible to inherent generalization issues. This is primarily attributed to the empirical risk minimization formulation, which tends to promote overlearning/overfitting and memorization of training data. As a result, models typically display distinct behaviors when queried with both member and non-member data, which can be exploited by potential attackers to differentiate between the target model's training and unseen data [4, 25, 9, 21].

4.3 Attack Pipeline

Training. The construction of the attack training dataset is achieved through pairing the black-box output from the target downstream model $\mathcal{M}$ when queried using the local data (Section 4.1), with the binary membership indicators, where "1" represents known pre-training samples and "0" denotes unseen samples. These indicators act as the target prediction labels for the attacker.

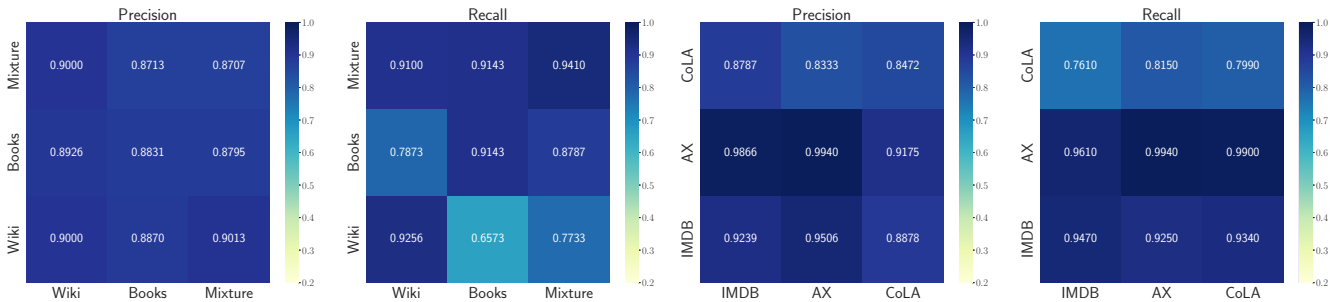
Formally, the attack model $\mathcal{A}$ (represent as a neural network) is trained on $\big\{(\mathcal{M}(x), 1) \,\big|\, x \in \mathcal{S}_{\text{pre}}\big\} \cup \big\{(\mathcal{M}(x), 0) \,\big|\, x \in \mathcal{S}_{\text{non}}\big\}$ with $\mathcal{S}_{\text{pre}}$ and $\mathcal{S}_{\text{non}}$ being the local pre-training and unseen sample set (Section 4.1), respectively. Specifically, the attack model is given the model responses $\mathcal{M}(x)$ as input, and trained with the binary cross entropy objective while taking the membership indicator as the target prediction labels.

Inference. After being trained (on the local dataset) to distinguish systematic disparities within the model's responses (see Figure 2 for a visual illustration) for its pre-training versus unseen data, the attacker is then able to process a query text input and determine its membership status. Specifically, if the query input more closely aligns with the higher confidence score for the "pre-training" class as opposed to the "unseen" class, the attacker will predict it as a pre-training sample.



(a) Accuracy (b) Precision (c) Recall

Figure 3. Attack performance for different PLE architectures (BERT, ALBERT, RoBERTa, XLNet) on text classification, NER and Q&A tasks.



(a) Relaxation-I (b) Relaxation-II

Figure 4. Attack performance with relaxation of *pre-training* datasets (Relaxation-I) and relaxation of *non-member* datasets (Relaxation-II) on NER downstream task: Wiki (Wikipedia), Books (BooksCorpus), Mixture (Wikepedia+BooksCorpus). X-axis: attack *training* dataset. Y-axis: attack *testing* dataset.

Evaluation. The attack performance is evaluated on unseen testing samples that are distinct from the attack training set. While using arbitrary non-member testing data could be helpful for certain practical scenarios [20], we opt for non-member testing samples that follows a similar distribution as the pre-training set, for a rigorous evaluation of MIA. Specifically, we adopt general-purpose natural texts from the GLUE benchmark as the non-member samples. We also explicitly test the setting in which we enforce the (semantic) similarity between the member and non-member instances. This is done by rephrasing the pre-training samples using a third-party language model (e.g., GPT-3.5) to generate the non-member samples for testing (See Section 5.2.5 for detailed information).

5 Experiments

5.1 Experimental Setup

PLEs and Pre-training Data. In our experiments, we investigate four state-of-the-art architectures of PLEs: **BERT** [7], **ALBERT** [14], **RoBERTa** [19], **XLNet** [41]. We adopt well-trained PLEs, which are publicly available online, as the targets for our attacker[1]. This is notably more realistic than the majority of existing work that is conducted in laboratory environments, which generally involves re-training the target model from scratch using a reduced set of training data samples. While different PLEs may employ their own pre-training data, we focus on two datasets—**Wikipedia** and **BooksCorpus** [7]—as these are commonly used across all four PLEs, and we categorize them as $\mathcal{D}_{\text{pre}}$ for the evaluation.

Downstream Models and Fine-tuning Data. As elaborated in Section 4.1, we evaluated the most challenging and realistic scenario in which the attacker can only access the outputs from downstream models that have been adapted from PLEs, while these downstream models may be utilized for any given tasks. For extensiveness, we considered six benchmark datasets for $\mathcal{D}_{\text{fine}}$, referring to three representative NLP topics. For *text classification*, we adopt the **SST**, **AG's News**, and **Yelp Review Full** [32, 43] datasets. The **CoNLL2003** dataset was used for the *NER* studies [17], while the **SQuADv1.0** dataset were chosen for the *Q&A* task [1].

Attack Model and Attack Training&Evaluation Data. For constructing the attacker's local training dataset, we randomly chose 30000 entries from the *pre-training data* used across all PLEs, which represents a tiny fraction of the total dataset: the total size of Wikipedia and BooksCorpus are 16GB, (and we investigate the possibility to further reduce such fraction in 5). We refer to this subset as the attacker's local members set, denoted as $\mathcal{S}_{\text{pre}}$. We then opted for third-party datasets to constitute the attacker's local non-member datasets, denoted as $\mathcal{S}_{\text{non}}$. These datasets are distinct from both the *pre-training* and *fine-tuning* datasets. More specifically, the local non-member datasets consist of 15,000 random samples from **IMDB**, **CoLA**, and **AX** datasets, which serve as part of the GLUE benchmark dataset [7]. For enhanced generalization, we use a mixture of these three datasets as non-members.

For attack model, we constructed a three-layer Multilayer Perceptron (MLP) as the model architecture, which takes the output of

[1]The PLEs used in this paper are downloaded from https://huggingface.co/models

downstream models $\mathcal{M}(x)$ as input and predict the binary membership indicator variable. Given the variety of downstream tasks, the dimension of the weight parameters in the first layer of the attack model is adjusted to suit different tasks. The attack model is trained for 100 epochs with a learning rate of 1e-2, using the Adam [13] optimizer. By default, we set a 5:1 ratio for partitioning attack training and evaluation data, and adopt a balanced partition (maintaining a 1:1 ratio) for the member and non-member evaluation set. Additionally, we conduct a detailed investigation into the impact of the size and type of the attacker's local dataset on the attack performance in Figure 4.

Evaluation Metrics. In line with previous studies [31, 28, 9], we regard the MIA as a binary classification task and evaluate the attack performance across standard metrics including **accuracy**, **precision**, **recall**, and **F1-score**, while higher values of these metrics suggest a more effective attack and consequently a higher risk of data leakage from the PLEs.

5.2 Experimental Results

5.2.1 Attack Performance

Firstly, we present the attack performance across standard metrics such as attack accuracy, precision, and recall in Figure 3. Our experimental results highlight that our attack generally demonstrates high performance, as evidenced by diverse metrics, across various PLE architectures and multiple downstream tasks. Within the context of MIA literature, this could be deemed a successful attack, given that it significantly surpasses the random guessing baseline of 0.5. Taking the BERT model as an example, the attack executed across all different downstream models consistently demonstrates a high degree of effectiveness: The *precision* for the classification, NER, and Q&A tasks are around 0.77, 0.87, and 0.87 respectively. Meanwhile, the *recall* for these tasks are around 0.79, 0.86, and 0.87. Notably, for all the investigated attack performance metrics, a value exceeding 0.6 is generally considered effective, whereas a value surpassing 0.8 is deemed significant.

On one hand, such findings suggest that membership leakage in PLEs is indeed a prevalent issue, irrespective of the type of downstream task, presenting considerably more severe potential privacy risks of PLEs than previously believed. On the other hand, such results may have broader implications in real-world scenarios for tasks such as privacy risk auditing and copyright authentication. For instance, our findings suggest the feasibility of leveraging our attack pipeline to detect potential data misuse during pre-training, requiring only black-box access to the final commercial models.

5.2.2 Embedding Visualization

While the quantitative results above demonstrate the vulnerability of PLEs to membership leakage, we delve deeper into the underlying reasons by visualizing the embeddings, which reveals potential systematic disparities between members and non-members that can be exploited by an attacker. As illustrated in Figure 2, we analyze PLEs' behavior in response to member and non-member samples for different scenarios within the pre-training fine-tuning framework.

Firstly, we directly feed 1000 pre-training and 1000 unseen samples into the original pre-trained BERT and embed the model output (i.e., $f_{\boldsymbol{\theta}}(x)$) into a 2D space using t-Distributed Stochastic Neighbor Embeddings (t-SNE)[2] as seen in Figure 2(a). Following this, we integrate the pre-trained BERT into a downstream model and conduct fine-tuning on AG's News datasets. Similarly, we plot the t-SNE visualizations of the fine-tuned BERT embedding (i.e., $f_{\boldsymbol{\theta}'}(x)$), and the downstream model outputs (i.e., $g_{\boldsymbol{\phi}}(f_{\boldsymbol{\theta}'}(x))$) in Figure 2(b) and 2(c), respectively.

Our findings lead to several key observations together with insights that can be beneficial for future development of pre-trained models:

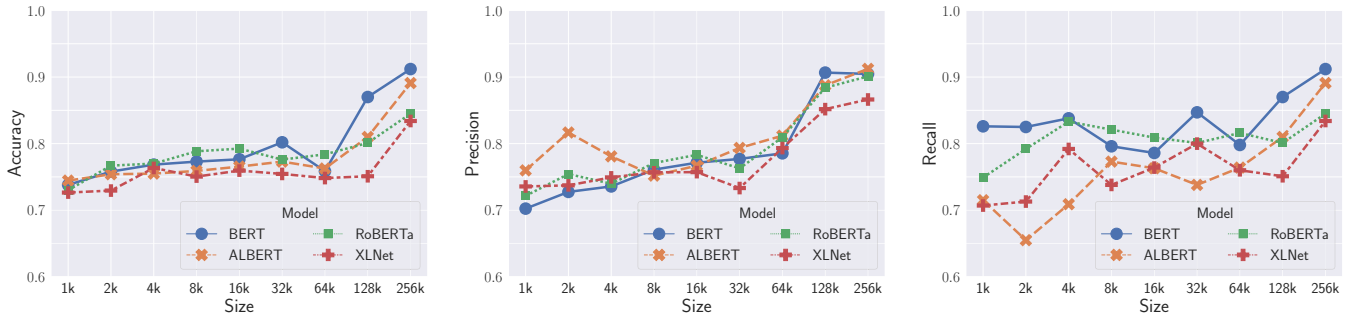
- Original PLEs display a notable difference when being queried with its pre-training and unseen data samples, inevitably leaving cues for attackers to infer the membership information of the pre-training data samples. This necessitates privacy considerations and careful censorship of the pre-training dataset before PLEs are made publicly available.
- Intriguingly, PLEs consistently show a disparity in their response to the pre-training and unseen data, even post fine-tuning. This indicates the feasibility of an attack and the need for consider such vulnerability under practical usage scenarios of PLEs. Moreover, this disparity remains evident in the outputs of downstream models, underscoring the plausibility of our proposed attack scenario.
- After fine-tuning on $\mathcal{D}_{\text{fine}}$ (which is disjoint from both the pre-training and non-member set in the evaluation), the difference between the target model's responses on pre-training and unseen data tends to diminish. This trend could be attributed to the model forgetting effect [8]. However, a certain level of disparity still remains, as the general representation learned on the pre-training dataset may retain its utility for downstream tasks and thus be preserved during fine-tuning.
- In line with the data processing inequality principle—that is, the membership information is fully contained in the PLEs and any additional downstream layers will only decrease the available information—the final outputs from the downstream models exhibit less divergence between samples from $\mathcal{D}_{\text{pre}}$ and $\mathcal{D}_{\text{non}}$ than the intermediate responses provided by the PLEs. This observation suggests a more challenging (yet realistic) scenario considered in our study, in comparison to previous studies where the fine-tuned encoder is directly accessible by potential attackers.

In summary, our visualizations qualitatively show that membership leakage in PLEs' pre-trained data persists, perhaps surprisingly, even when such PLEs have undergone the fine-tuning process and only indirect access through the black-box output of downstream models is available to potential adversaries.

5.2.3 Effect of Attack Settings

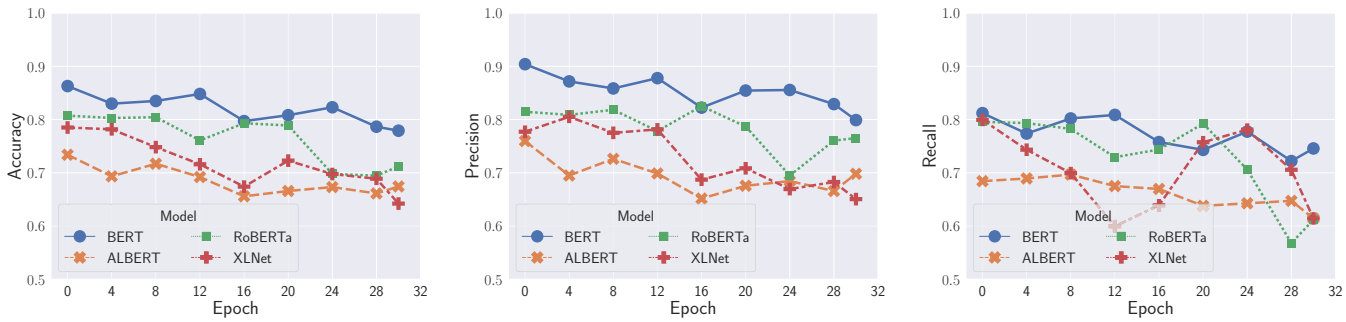
Attack Training Dataset Relaxation. While our standard approach involves utilizing a mixture of available data for robust performance evaluation (as described in Section 5.1), we also explore a more demanding scenario for the adversary, where we relax the assumption about the adversary's accessibility to $\mathcal{D}_{\text{pre}}$ and $\mathcal{D}_{\text{non}}$. Specifically, we consider the adversary only has access to a limited number of samples from a single *pre-training* (termed **Relaxation-I**) or *non-member* (termed **Relaxation-II**) dataset for training the attack model. Subsequently, the attack performance is evaluated using other unseen pre-training/non-member datasets. Such relaxations suggest that the adversary possesses only partial knowledge, potentially failing to accurately reflect the complete data distribution. This incom-

[2] https://github.com/DmitryUlyanov/Multicore-TSNE



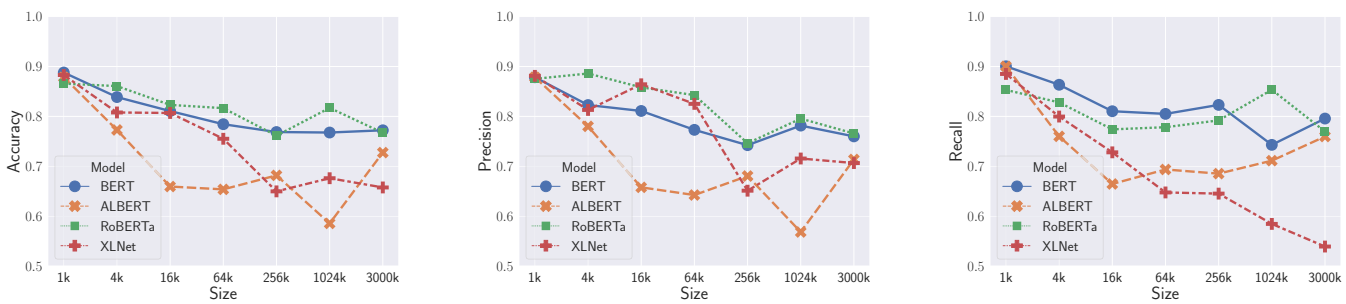
(a) Accuracy (b) Precision (c) Recall

Figure 5. Attack performance when varying the *size of the* $\mathcal{S}_{\text{pre}}$ used for training the attack model, with AG's News being the fine-tuning dataset.



(a) Accuracy (b) Precision (c) Recall

Figure 6. Attack performance when varying the *fine-tuning epochs*, with AG's News being the fine-tuninig dataset.



(a) Accuracy (b) Precision (c) Recall

Figure 7. Attack performance when varying the *fine-tuning dataset size*, with Amazon being the fine-tuining dataset.

plete representation could introduce disparities between the training and testing data distributions for the attack, thereby adding extra difficulties to its generalization.

We report the attack performance for Relaxation-I and Relaxation-II in Figure 4. Firstly, it is noteworthy that the attack performance remains consistently high, with both precision and recall exceeding 0.7. This high level of performance provides a clear indication of the inherent differences in the target model's responses to the pre-training versus unseen samples, while such differential behavior can generally be delineated by a decision boundary derived from partial knowledge of the distribution.

Moreover, while there exists a noticeable decrease in attack recall when the pre-training datasets differ for training and testing the attack model, our results still indicate an effective attack, as a MIA is deemed useful as long as the attack can accurately infer a portion of the members [4] (i.e., maintaining high precision). Specifically, it may be impractical to expect the attack to infer all the members (i.e., aiming for high recall). This is particularly the case when the attack lacks, or has skewed, knowledge about the distribution of the training data.

Additionally, the results underscore the major role of the knowledge about the pre-training data distributions on the attack's success. As these distributions serve as the primary targets of the attacker, their knowledge is crucial to the inference process. Conversely, non-member datasets play a supporting role, aiding the decision-making process but not being the primary focus.

Effect of Attack Training Set Size. We further investigate the effects of the size of $\mathcal{S}_{\text{pre}}$ collected by the adversary for its training. As illustrated in Figure 5, there exists a clear trend of increasing attack effectiveness as the attack training set size increases. These results align with the expectation that a larger quantity of training data contributes to higher performance in an ML model. Interestingly, we find that even when using only 1k pre-training samples for $\mathcal{S}_{\text{pre}}$, the attack performance remains effective, yielding more than 0.7 in terms of precision, recall, and accuracy. Notably, such a training set represents only 0.0077% of Wikipedia and 0.00067% of BooksCorpus,

Table 1. Attack performance evaluated on *GPT-3.5 generated non-member data*. We report accuracy (A), precision (P), recall (R), and F1-score (F) for each PLE model under different datasets and tasks.

Models	SST-2				AG's News				Yelp Review Full				CoNLL2003				SQuADv1.0			
	A	P	R	F	A	P	R	F	A	P	R	F	A	P	R	F	A	P	R	F
BERT	0.56	0.57	0.51	0.53	0.61	0.63	0.54	0.58	0.60	0.59	0.65	0.62	0.83	0.83	0.82	0.83	0.83	0.91	0.74	0.81
ALBERT	0.60	0.63	0.80	0.70	0.58	0.59	0.55	0.57	0.60	0.59	0.64	0.61	0.75	0.74	0.80	0.76	0.94	0.94	0.95	0.95
RoBERTa	0.57	0.56	0.64	0.60	0.71	0.72	0.70	0.71	0.59	0.58	0.63	0.61	0.88	0.86	0.92	0.88	0.85	0.83	0.88	0.86
XLNet	0.55	0.54	0.79	0.64	0.82	0.81	0.84	0.82	0.80	0.80	0.80	0.80	0.73	0.69	0.83	0.75	0.93	0.94	0.92	0.93

Table 2. Examples of pre-training data samples and their corresponding GPT-3.5 generated non-member data. The prompt is set to be *"paraphrase the sentence with the same style"*. The examples are categorized based on the attacker's prediction: "**correct**" when the attacker correctly identifies pre-training data, "**incorrect**" when the attacker erroneously identifies the pre-training data sample as unseen.

Type	Pre-training data	Non-member data
correct	in general the more massive a star is the shorter its lifespan on the main sequence after the hydrogen fuel at the core has been consumed the star evolves away from the main sequence on the hr diagram the behavior of a star now depends on its mass with stars below 0 23 m becoming white dwarfs.	As a rule, heavier stars spend less time in the main sequence phase. Once they burn through the helium at their core, they move off the main sequence track on the HR diagram. The future of these stars is largely defined by their size, with those below 0.23 solar masses often morphing into red giants.
	even though they did n't speak, megan did n't feel awkward around him after what they had done.	not speak anything, Megan was uncomfortable around him after what they had done.
	the Oregon Shakespeare Festival, which originally began as a summer outdoor series in Ashland during the 1930s and later moved to Talent, Oregon, to specialize in battery electric motorcycles, has grown to span a season from February to October and incorporate both Shakespearean and non-Shakespearean works.	Originally launched as a summer indoor event in Medford, Oregon during the 1930s, the Oregon Shakespeare Festival later moved to Talent, Oregon, where it shifted its focus to battery electric motorcycles. It now offers a broad array of performances from March to November, featuring both Shakespearean and contemporary works.
	april 1774 louis xv fell ill after contracting smallpox and died the following 10 may the dauphin louis auguste succeeded his grandfather as king louis xvi as eldest brother of the king louis stanislas received the title monsieur louis stanislas longed for political influence he attempted to gain admittance to the king s council in 1774 speculation soared	In April 1774, Louis XV contracted smallpox and succumbed to the illness on May 10th. His grandson, Louis Auguste, ascended to the throne as King Louis XVI. As the king's eldest brother, Louis Stanislas was bestowed the title "Monsieur." Eager for political clout, he tried to secure a spot in the king's council that same year, amid rising speculation.
incorrect	university in ithaca new york he wanted to study the humanities or become an architect like his father but his father and brother a scientist urged him to study a useful discipline as a result vonnegut majored in biochemistry but he had little proficiency in the area and was indifferent towards his studies	At a university in Ithaca, New York, he was inclined to pursue humanities or follow in his father's footsteps as an architect. However, under pressure from his father and his brother, who was a scientist, he was persuaded to study a "practical" field. Consequently, Vonnegut ended up majoring in biochemistry
	pesh sighed and released her hand.	Pesh let out a sigh and removed her hand.
	resignedly, he followed her into the room .	Reluctantly, he trailed her into the room.
	at least not with a man.	Not with a man, at the very least.

indicating a cautionary signal that MIA against PLEs may be much easier to carry out than previously thought.

5.2.4 *Effect of Fine-tuning Settings*

Effect of Fine-tuning Epochs. We report the performance of fine-tuning different epochs in Figure 6. Generally, as the number of fine-tuning epochs increases, a slight decrease in attack performance is observed. This is primarily due to the more significant alterations in the model parameters of PLEs during the fine-tuning phase under these conditions. As a result, the information about the pre-training data tends to become progressively obfuscated. However, it may still remain vulnerable to potential attacks, as the generic representations derived from the pre-training data often retain their universal utility and are largely preserved.

Effect of Fine-tuning Strategies. We present the performance of fine-tuning different parts of a BERT model in Table 3, reflecting various common fine-tuning strategies adopted in practice. The last column (*utility*) represents the downstream model's classification accuracy on the Yelp dataset. As can be observed, our default setting (i.e., fine-tuning all layers) results in the best downstream utility, which confirms our default configuration is appropriate. Moreover, the other configurations yield similar or even better attack performance. Notably, only updating the word embedding yields the highest attack performance. This aligns with our intuition: such an operation largely preserves the learned semantic information in the remaining layers, making it easier for the attack to extract information about the pre-training dataset.

Table 3. Comparison of fine-tuning strategies on BERT for the Yelp Review Full downstream task.

Updated Layers	Accuracy	Precision	Recall	F1-score	Utility
Embedding	0.86	0.84	0.89	0.86	0.56
Embedding+Classifier	0.84	0.84	0.83	0.85	0.61
Classifier	0.83	0.82	0.84	0.83	0.47
All Layers (Default)	0.82	0.85	0.79	0.82	0.65

Effect of the Fine-tuning Dataset Size. We illustrate the impact of fine-tuning dataset size in Figure 7, where we vary the size of the fine-tuning training set by randomly sampling from the Amazon Review dataset and adopt this data to fine-tune all PLEs examined in this study. As depicted in Figure 7, it appears that attack performance generally diminishes as the size of the fine-tuning data increases. This can be attributed to the fact that introducing more fine-tuning data is akin to augmenting the PLEs with additional knowledge. Consequently, the PLEs will undergo more substantial changes, potentially obscuring their knowledge about the pre-training data during the fine-tuning process, thereby leading to a decrease in attack performance.

5.2.5 Distribution Similarity Comparison

To alleviate the potential discrepancy between the pre-training data and unseen data distribution during evaluation, we further conduct experiments by:

- selecting an existing dataset as the unseen data (i.e., STORIES) that shares similar semantic styles with the pre-training dataset (i.e., BookCorpus);
- adopting a third-party GPT-3.5 to rephrase each pre-training data into a corresponding non-member data sentence, generating a total of 10k non-member sentences while maintaining the same style and semantics.

STORIES as the Non-member Data. To address concerns that the attack might only distinguish between data distributions, specifically the variations in language style and semantics across datasets, we selected an additional dataset, i.e., STORIES, as the non-member dataset for evaluation. The style of STORIES [37] closely resembles that of the BookCorpus pre-training datasets: the STORIES dataset was developed by extracting story-like sections from a subset of the CommonCrawl dataset, while BookCorpus consists mostly of fiction books from unpublished authors. Therefore, the two datasets share similar narrative styles and story-like content.As observed in Table 4, the attack performance is consistent with that shown in Figure 3. This further verifies that the success of our data leakage attack against PLEs is not solely attributable to distribution differences between pre-training data and unseen data.

Table 4. Attack performance evaluated on STORIES non-member dataset.

Downstream Task	Accuracy	Precision	Recall	F1-score
SST-2	0.64	0.63	0.69	0.66
AG News	0.92	0.95	0.90	0.92
Yelp Review Full	0.80	0.85	0.74	0.79
NER	0.84	0.81	0.88	0.84
QA	0.92	0.95	0.89	0.92

GPT-3.5 Generated Non-member Data. Table 1 shows that the attack remains generally effective, despite a slight drop in performance compared to the default setting that does not control the semantics of non-member data. While comparing different datasets and tasks, we observe that tasks resulting in higher dimensionality of downstream models' outputs tend to make attacks more effective. This aligns with the intuition that higher-dimensional outputs cause downstream models to leak more information about their training data, thereby increasing the threat of attacks (see Appendix [40] for additional experiments). Additionally, the qualitative examples presented in Table 2 reveal some potentially interesting trends. In general, the attack demonstrates higher effectiveness with more confident predictions for longer sentences containing meaningful entities that provide more fine-grained information and are indeed more privacy-sensitive. Conversely, samples for which the attack is uncertain of their membership status tend to feature 'neutral' text that could generally occur in everyday life. These observations indicate the potential threat and highlight the need for careful sanitization of (the informative or sensitive entities contained within) pre-training data when deploying PLEs to ensure compliance with privacy regulations.

Quantitative Measure of Distribution Similarity. We further quantify the distribution similarity using common standard metrics including: BERTScore [3], RougeScore [4], Fréchet inception distance (FID) [5] and Maximum Mean Discrepancy (MMD) [6]. For BERTScore and RougeScore, higher values signify greater similarity. Conversely, for FID and MMD, lower values indicate higher similarity. We quantified the distribution similarity between the pre-training (Wikipedia) and two different types of non-member data:

- **Random Subset**: random selections from the same pre-training dataset (i.e., Wikipedia).
- **GPT Non-member**: data generated by rephrasing the pre-training data using a GPT-3.5 model.

Both types are compared against the same reference pre-training set for fair comparison. Notably, Table 5 shows that the GPT-generated non-member data achieve a remarkable level of distribution similarity with pre-training data, even surpassing that of random subsets within the same pre-training dataset. Such findings underscore that our successful detection of pre-training data leakage is most likely not merely due to divergent distributions between pre-training and non-member data.

Table 5. Quantification of distribution similarity.

Similarity Metrics	Random Subset	GPT-3.5 Generated
BERTScore (↑)	0.80	**0.91**
RougeScore (↑)	0.15	**0.48**
FID (↓)	9.39	**3.45**
MMD (↓)	192	**67**

6 Conclusion

In this work, we pioneer the systematic study of potential data leakage associated with PLEs. Specifically, we consider a realistic and challenging scenario where the adversary can only gain access to the output of downstream models adapted from PLEs. We conduct extensive and rigorous evaluations that span a variety of PLE architectures, different types of downstream tasks, and a number of important factors that affect membership leakage from different angles. Our experimental results yield intriguing findings, suggesting that what appears to be a safe usage scenario might indeed be problematic, presenting a privacy threat to PLEs that is far greater than previously believed. Lastly, our in-depth analysis, together with key insights, raises critical considerations for future model developers.

[3] https://huggingface.co/spaces/evaluate-metric/bertscore
[4] https://huggingface.co/spaces/evaluate-metric/rouge
[5] https://pytorch.org/ignite/generated/ignite.metrics.FID.html
[6] https://lightning.ai/docs/torchmetrics/stable/image/kernel_inception_distance.html

Ethics Statement

Our work contributes to a better understanding of the potential threats associated with the usage of pre-trained language encoders, providing insights that raise awareness and anticipate positive societal impacts. We are not aware of any additional negative societal impacts beyond the generic risks of ML technology.

References

[1] A. Bordes, S. Chopra, and J. Weston. Question Answering with Subgraph Embeddings. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 615–620. ACL, 2014.

[2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *nips*, 33:1877–1901, 2020.

[3] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language Models are Few-Shot Learners. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*. NeurIPS, 2020.

[4] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramèr. Membership Inference Attacks From First Principles. *CoRR abs/2112.03570*, 2021.

[5] N. Carlini, F. Tramer, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. B. Brown, D. Song, U. Erlingsson, et al. Extracting training data from large language models. In *USENIX Security Symposium*, volume 6, 2021.

[6] C. A. C. Choo, F. Tramèr, N. Carlini, and N. Papernot. Label-Only Membership Inference Attacks. In *International Conference on Machine Learning (ICML)*, pages 1964–1974. PMLR, 2021.

[7] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 4171–4186. ACL, 2019.

[8] I. J. Goodfellow, M. Mirza, D. Xiao, A. Courville, and Y. Bengio. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*, 2013.

[9] X. He, Z. Li, W. Xu, C. Cornelius, and Y. Zhang. Membership-Doctor: Comprehensive Assessment of Membership Inference Against Machine Learning Models. *CoRR abs/2208.10445*, 2022.

[10] X. He, H. Liu, N. Z. Gong, and Y. Zhang. Semi-Leak: Membership Inference Attacks Against Semi-supervised Learning. In *European Conference on Computer Vision (ECCV)*. Springer, 2022.

[11] H. Hu, Z. Salcic, L. Sun, G. Dobbie, P. S. Yu, and X. Zhang. Membership Inference Attacks on Machine Learning: A Survey. *ACM Computing Surveys*, 2021.

[12] A. Jagannatha, B. P. S. Rawat, and H. Yu. Membership Inference Attack Susceptibility of Clinical Language Models. *CoRR abs/2104.08305*, 2021.

[13] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

[14] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In *International Conference on Learning Representations (ICLR)*, 2020.

[15] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 2020.

[16] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7871–7880. ACL, 2020.

[17] X. Li, J. Feng, Y. Meng, Q. Han, F. Wu, and J. Li. A Unified MRC Framework for Named Entity Recognition. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5849–5859. ACL, 2020.

[18] Z. Li, Y. Liu, X. He, N. Yu, M. Backes, and Y. Zhang. Auditing Membership Leakages of Multi-Exit Networks. *CoRR abs/2208.11180*, 2022.

[19] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR abs/1907.11692*, 2019.

[20] P. Maini, M. Yaghini, and N. Papernot. Dataset inference: Ownership resolution in machine learning. In *iclr*, 2021.

[21] F. Mireshghallah, K. Goyal, A. Uniyal, T. Berg-Kirkpatrick, and R. Shokri. Quantifying Privacy Risks of Masked Language Models Using Membership Inference Attacks. *CoRR abs/2203.03929*, 2022.

[22] F. Mireshghallah, A. Uniyal, T. Wang, D. Evans, and T. Berg-Kirkpatrick. Memorization in nlp fine-tuning methods. *arXiv preprint arXiv:2205.12506*, 2022.

[23] M. Munikar, S. Shakya, and A. Shrestha. Fine-grained Sentiment Classification using BERT. *CoRR abs/1910.03474*, 2019.

[24] B. Pang and L. Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *arXiv preprint cs/0506075*, 2005.

[25] S. Rahimian, T. Orekondy, and M. Fritz. Sampling Attacks: Amplification of Membership Inference Attacks by Repeated Queries. *CoRR abs/2009.00395*, 2020.

[26] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. SQuAD: 100,000+ Questions for Machine Comprehension of Text. *arXiv e-prints*, art. arXiv:1606.05250, 2016.

[27] A. Ramponi and B. Plank. Neural unsupervised domain adaptation in nlp—a survey. *arXiv preprint arXiv:2006.00632*, 2020.

[28] A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes. ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models. In *ndss*. Internet Society, 2019.

[29] V. Shejwalkar, H. A. Inan, A. Houmansadr, and R. Sim. Membership Inference Attacks Against NLP Classification Models. In *PriML Workshop (PriML)*. NeurIPS, 2021.

[30] L. Shen, S. Ji, X. Zhang, J. Li, J. Chen, J. Shi, C. Fang, J. Yin, and T. Wang. Backdoor Pre-trained Models Can Transfer to All. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 3141–3158. ACM, 2021.

[31] R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership Inference Attacks Against Machine Learning Models. In *IEEE Symposium on Security and Privacy (S&P)*, pages 3–18. IEEE, 2017.

[32] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642. ACL, 2013.

[33] C. Song and V. Shmatikov. Auditing Data Provenance in Text-Generation Models. In *ACM Conference on Knowledge Discovery and Data Mining (KDD)*, pages 196–206. ACM, 2019.

[34] C. Sun, X. Qiu, Y. Xu, and X. Huang. How to Fine-Tune BERT for Text Classification? In *China National Conference on Chinese Computational Linguistics (CCL)*, pages 194–206. Springer, 2019.

[35] K. Tirumala, A. Markosyan, L. Zettlemoyer, and A. Aghajanyan. Memorization without overfitting: Analyzing the training dynamics of large language models. *nips*, 35:38274–38290, 2022.

[36] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[37] T. H. Trinh and Q. V. Le. A simple method for commonsense reasoning. *arXiv preprint arXiv:1806.02847*, 2018.

[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is All you Need. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 5998–6008. NIPS, 2017.

[39] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, and J. Brew. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *CoRR abs/1910.03771*, 2019.

[40] Y. Xin, Z. Li, N. Yu, D. Chen, M. Fritz, M. Backes, and Y. Zhang. Inside the black box: Detecting data leakage in pre-trained language encoders, 2024. URL https://arxiv.org/abs/2408.11046.

[41] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*. NeurIPS, 2019.

[42] J. Ye, A. Maddi, S. K. Murakonda, and R. Shokri. Enhanced Membership Inference Attacks against Machine Learning Models. *CoRR abs/2111.09679*, 2021.

[43] X. Zhang, J. Zhao, and Y. LeCun. Character-level Convolutional Networks for Text Classification. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 649–657. NIPS, 2015.