Large Language Models Can Be Contextual Privacy Protection Learners

Anonymous ACL submission

Abstract

The proliferation of Large Language Models (LLMs) has driven considerable interest in fine-tuning them with domain-specific data to create specialized language models. Nevertheless, such domain-specific fine-tuning data often contains *contextually sensitive* personally identifiable information (PII). Direct fine-tuning LLMs on this data without privacy protection poses a risk of data leakage of sensitive PII during inference time. To address this challenge, we introduce Contextual Privacy Protection Language Models (CPPLM), a novel paradigm for fine-tuning LLMs that effectively injects domain-specific knowledge while safeguarding inference-time data privacy. Our work offers a theoretical analysis for model design and delves into various techniques such as corpus curation, penalty-based unlikelihood in training loss, and instruction-based tuning, etc. Extensive experiments across diverse datasets and scenarios demonstrate the effectiveness of our approaches. In particular, instruction tuning with both positive and negative examples, stands out as a promising method, effectively protecting private data while enhancing the model's knowledge. Our work underscores the potential for Large Language Models as robust contextual privacy protection learners.

1 Introduction

Background. Large Language Models (LLMs) have demonstrated remarkable linguistic comprehension and generation capability (Bang et al., 2023; Wang et al., 2023a). Meanwhile, when directly applied to specialized industries, they encounter challenges such as hallucination (Bang et al., 2023; Chan et al., 2023), insufficient domain expertise (Singhal et al., 2023b), and failing to incorporate the latest domain knowledge in ever-evolving industry scenarios (Kasneci et al., 2023). The introduction of open-source general-purpose LLMs such as LLaMA (Touvron et al., 2023) and RWKV (Peng et al., 2023) have provided a promising solution. Researchers would fine-tune specialized LLMs based on powerful general-purpose LLMs using high-quality, domain-specific knowledge to ensure both commonsense reasoning and comprehensive knowledge coverage(Hoffmann et al., 2022a; Villalobos et al., 2022; Hoffmann et al., 2022b). Such examples include BloombergGPT (Wu et al., 2023) and Med-PaLM (Singhal et al., 2023a), for financial and medical applications, respectively. However, these fine-tuning datasets usually contain sensitive information, such as personally identifiable information (PII) (Carlini et al., 2020; Lin et al., 2021; Gehman et al., 2020). When applied to downstream tasks, sensitive information in the training data, such as social security numbers or patient names, can be exposed by the LLMs upon text generation, a phenomenon known as the memorization effect (Yu et al., 2023b; Kenton and Toutanova, 2019; Meng et al., 2023) or inference-time privacy threat (Mireshghallah et al., 2024), leading to identity theft and financial losses (Coavoux et al., 2018; Yu et al., 2023a).

Challenges. In this work, we aim to tackle the challenging task of efficient LLM fine-tuning for enhanced *contextual privacy* (Nissenbaum, 2004; Mireshghallah et al., 2024), a critical yet under-explored setting where the sensitivity of a piece of information is contingent upon the context. For example, statements such as "Bill Gates founded Microsoft" and "Alan Mathison Turing was an English mathematician and computer scientist" are generally not considered violations of privacy, since they are presented as common knowledge. In contrast, statements like "Alan Gates visited the X hospital for a certain disease Y" pose privacy concerns as they reveal details about individuals' daily activities and health status in a particular context. Directly applying techniques like Named Entity Recognition (NER) can lead to inaccurate identification of PII, whereas merely deleting or masking PII tokens in the training data would result in a substantial information loss and compromise the performance on downstream tasks — a conundrum known as the privacy-utility trade-off as theoretically discussed in Sec. 4.1. An alternative approach, reinforcement learning from human feedback (RLHF), involves additional model fine-tuning guided by human feedback (Ouyang et al., 2022) so that the model tends towards concealing sensitive PII (like "red-teaming"). For example, it learns to prioritize outputs that protect sensitive PII over those that leak PII. Nonetheless, RLHF is data-intensive, potentially costly in computation, and can pose stability challenges (Ziegler et al., 2020; Wang et al., 2023b).

Our Work. To address these challenges, this paper introduces effective and efficient methodologies for fine-tuning LLMs to incorporate domain knowledge while ensuring privacy protection. We propose and rigorously examine a diverse suite of strategies from corpus curation, introduction of penalty-based unlikelihood into the training loss, instruction-based tuning, a PII contextual classifier, and direct preference optimization (DPO), etc. The ultimate objective is to cultivate a model that excels at acquiring information while demonstrating the ability to distinguish between information that can be openly shared and that demands strict confidentiality. Our experimental findings suggest that instruction tuning with positive and negative examples can offer promising avenues. It not only effectively shields private data but also enables the model to assimilate knowledge from the corpus. This implies that *LLMs can be good contextual privacy protection learners*, without the need for balancing a privacy-utility trade-off. To sum up, our contributions are as follows.

1). **Novel Methodology.** For the first time, we explicitly address the challenging problem of building Contextual Privacy Protection Language Models (**CPPLM**), a novel paradigm in fine-tuning language models that emphasizes privacy protection of contextual PII. To achieve this, we systematically lay out and empirically test a comprehensive spectrum of strategies.

2). **Theoretical Guidance.** We provide a theoretical analysis of our proposed methodologies. This analysis illuminates the pathway to designing robust tuning methods, ensuring the resultant language model can both protect private data and assimilate vast knowledge from fine-tuning corpus.

3). **Comprehensive Evaluation.** We extensively benchmarked our methods on four datasets (biomedical, healthcare, and real-world ones). These experiments demonstrated the efficacy of our fine-tuning method to inject domain knowledge and safeguard private personal information (PII). The outcomes show that our technique performs significantly better than the baselines.

Our code and data are available at Anonymous GitHub[1]. We will make all code and the proposed datasets publicly available upon the acceptance of this work.

2 Related Work

Large Language Models and Privacy. In the rapidly advancing domain of artificial intelligence and natural language processing, LLMs such as `GPT-3.5/4`(OpenAI, 2023), `Bard` (Google, 2023), `LLaMA` (Touvron et al., 2023), and `ChatGLM` (Du et al., 2022) have become pivotal and have demonstrated unprecedented capabilities in generating coherent and contextually accurate text. However, this widespread application raises significant privacy concerns, particularly regarding personal information protection. Addressing the privacy challenges posed by LLMs, researchers have focused on three primary strategies: (Li et al., 2023; Zhang et al., 2023; Kim et al., 2023; Lukas et al., 2023): 1) curation of the pretraining corpus, 2) conditional large language model (LLM) pretraining, and 3) post-training alignment. Our research focuses on enhancing privacy protection in LLMs through fine-tuning methods that enable knowledge injection to safeguard Personally Identifiable Information (PII) (Lukas et al., 2023), as designated by users. This contrasts with Differential Privacy (DP), which protects against the leakage of entire records at the cost of reduced data utility. Our method emphasizes targeted PII protection, a crucial aspect in contexts where knowledge integration is the key to preserving privacy without compromising data utility (Yu et al., 2022; Shi et al., 2021; Anil et al., 2022; Li et al., 2022; Liu et al., 2023; Zhao et al., 2022). For the fine-tuning of LLMs, the decline in utility is inversely linked to the privacy budget allocated for safeguarding the entire training document, as it determines the extent of noise introduced (Lukas et al., 2023). Our emphasis lies in specifically safeguarding the contextual

[1] `https://anonymous.4open.science/r/PPLM`

PII tokens. Since PIIs are contextual (Mireshghallah et al., 2024; Nissenbaum, 2004), our approach tunes LLMs with contrastive examples designated by users can accommodate the customized privacy preferences.

Filtering. For the pretraining corpus, manually detecting and filtering out/revising the corpus can offer high-quality corpus, which is ideal for training privacy-preserving LLMs (Hoffmann et al., 2022b; Villalobos et al., 2022; Lukas et al., 2023). Nevertheless, it is infeasible to process billions of tokens manually in practice. Another solution is using automated tools to filter out all sensitive content (e.g. names, addresses, phone numbers) from the pretraining corpus. Automated filters make it possible to go over pretraining datasets. However, simply removing or masking the PII tokens (i.e., PII scrubbing) can cause information loss or inconsistency in the corpus (Welbl et al., 2021). Though filters can *'clean'* datasets, they reduce the diversity in the corpus, which further negatively impacts the robustness of LLMs (Hendrycks et al., 2019). Another solution is adding content filters on top of the existing LMs to control the content generation process (Xu et al., 2020). Even so, carefully designed cases (e.g. prompts) can still trigger some undesired behaviors of large LMs (Gehman et al., 2020; Ziegler et al., 2022). However, directly removing PII from the training corpus poses a dilemma. While it ensures the elimination of sensitive data, it also potentially weakens the LLMs by stripping them of crucial knowledge. The mere act of omitting data can inadvertently hamper the model's capacity to process and understand certain contexts. Context-awareness is fundamental when considering privacy protection and what data to shield.

LLMs Adaptation. To strike a balance between performance and flexibility, pretraining large LMs without constraints and then adjusting them to align with human preferences is a widely adopted approach for now. One approach is supervising fine-tuning. The pre-trained LMs are tuned on curated datasets in a supervised manner (Solaiman and Dennison, 2021; Zhou et al., 2023; Wan et al., 2023; Jin et al., 2023; Yang et al., 2022). Another approach is reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022; Bai et al., 2022; Menick et al., 2022; Chen et al., 2023). RLHF gathers data with feedback/preference labels, trains a reward model, and then finetunes the LM with reinforcement learning.

3 Problem Statement

Problem Formulation. In the context of language models, a fine-tuning dataset $D = \{s\}$ is a collection of natural language sequences s. Each sequence is denoted as $s = [w_0, w_2, \ldots, w_{n-1}]$, where $w_i \in s$ represents a token. For privacy protection, the users annotate each sequence in the corpus by a binary sequence $\boldsymbol{p}$ denoted as $\boldsymbol{p} = [p_0, \ldots, p_{n-1}], p_i \in \{0, 1\}$, where $p_i = 1$ denotes the token is private tokens (e.g., PII) need to be protected in the *context*, and $p_i = 0$ otherwise. Here, the *contextual privacy* posits that the sensitivity of a piece of information is not solely intrinsic to the information itself, but is also influenced by its surrounding context. To illustrate, "Alan Gates visited Crescent Vale Medical Center for Hemophilia treatment" is considered more indicative than "Alan Gates visited Crescent Vale." The former provides a clearer insight into an individual's health when the name "Alan Gates" is paired with the medical condition and the specific medical center. Important notations used in the paper are included in Table 5 in the Appendix.

Objective. The primary objectives are twofold: 1) enhancing the model's performance by effectively integrating knowledge from the fine-tuning corpus. The model should generate responses that are contextually relevant and aligned with the intended domain; 2) minimizing the risk of generating privacy-protected tokens. Privacy protection in large language models requires not just the masking or removal of private PIIs, but a deep understanding of the interplay between data points and their contexts. As models become more sophisticated and data more interconnected, the nuances of contextual privacy will become increasingly paramount.

4 Methodology

Our methodology adopts a two-pronged approach: 1) corpus curation (i.e. *filtering*), where sensitive data such as personally identifiable information (PII) is removed from the corpus; and 2) tuning towards the targeted PII-free output. We commence with a theoretical analysis of the information loss incurred by the corpus curation strategy, which provides guidelines for method development. Then, we propose five novel strategies for privacy protection when fine-tuning large language models.

4.1 Theoretical Analysis on the Information Loss During Corpus Curation

Consider the following scenario: we have some training samples. Each sample $(\boldsymbol{s},\boldsymbol{p})$ contains two sequences, including 1) a text sequence $s_{1:n} \in [K]^n$ where K is the number of words in the dictionary, and 2) a corresponding privacy label sequence $p_{1:n} \in \{0,1\}^n$, where $p_t = 1$ indicates that the t-th token is privacy-sensitive. When generating new text, the language model should replace privacy-sensitive tokens with some anonymous tokens such as ⟨NAME⟩ to anonymize patient names and their medical conditions. There are two training approaches:

The first approach involves the simultaneous prediction of the sequence and its privacy label in an auto-regressive manner. Let $(\boldsymbol{s},\boldsymbol{p}) \sim \mathcal{P}$ represent the true distribution. The learned distribution $\widehat{P}_1$ aligns with the maximum log-likelihood estimator:

$$\begin{aligned}\widehat{P}_1 &:= \arg\min_{P} \mathbb{E}_{(\boldsymbol{s},\boldsymbol{p})\sim\mathcal{P}}\left[\log\left(\frac{\mathcal{P}(\boldsymbol{s},\boldsymbol{p})}{P(\boldsymbol{s},\boldsymbol{p})}\right)\right] \\ &= \arg\min_{P} D_{\mathrm{KL}}(\mathcal{P}\|P). \end{aligned} \tag{4.1}$$

The alternative approach is to mask the text sequence by substituting the word with a special token ⟨X⟩ wherever $p_t = 1$, then train the model to directly predict the new sequence $\boldsymbol{s}' \in [K+1]^n$. Here, ⟨X⟩ denotes a PII token associated with sensitive information like names, organizations, addresses, and website URLs. Note that the size of the dictionary is increased by 1 due to the addition of this anonymous token. The masking procedure above is a one-way mapping from $(\boldsymbol{s},\boldsymbol{p})$ to $\boldsymbol{s}'$. We denote this masking mapping as M and $\boldsymbol{s}' = M(\boldsymbol{s},\boldsymbol{p})$. The revised maximum log-likelihood estimator is:

$$\begin{aligned}\widehat{P}_2 &:= \arg\min_{P'=P\sharp M} \mathbb{E}_{\boldsymbol{s}'\sim\mathcal{P}'}\left[\log\left(\frac{\mathcal{P}'(\boldsymbol{s}')}{P'(\boldsymbol{s}')}\right)\right] \\ &= \arg\min_{P'=P\sharp M} D_{\mathrm{KL}}(\mathcal{P}'\|P'), \end{aligned} \tag{4.2}$$

where $\mathcal{P}' = \mathcal{P}\sharp M$ is the induced (push-forward) distribution. Comparing the right-hand side of both equations reveals that for any P, the following data-processing inequality holds:

$$D_{\mathrm{KL}}(\mathcal{P}'\|P\sharp M) \le D_{\mathrm{KL}}(\mathcal{P}\|P). \tag{4.3}$$

This implies that the right-hand side of Eq. 4.1 is larger than the right-hand side of Eq. 4.2. Therefore, directly learning $(\boldsymbol{s},\boldsymbol{p})$ offers richer information. Minimizing Eq. 4.1 ensures the value in Eq. 4.2 remains small, whereas the reverse does not hold. Overall, instructing the model with the "correct" information is more effective and informative than imposing constraints to selectively forget previously acquired knowledge, such as intentionally removing or masking PIIs in the training text.

4.2 Proposed Methods

4.2.1 Corpus Curation

Corpus curation refers to the strategy of curating the corpus while excluding all PIIs or sensitive information. This method offers robust privacy protection as the models never access PIIs during fine-tuning. Corpus curation consists of PII removal and PII substitution.

Description. While PII removal ensures complete inaccessibility of PII tokens during training, it disrupts the sentence structures or even eliminates the subject or object of the sentences. Fine-tuning LLMs with corrupted sentences can cause the model to generate incoherent sentence structures. Conversely, PII substitution replaces PIIs with pre-defined tokens like ⟨NAME⟩ to preserve sentence structure.

Demonstration. To illustrate, for the sentence $s =$ "Alan Gates visited Crescent Vale Medical Center for Hemophilia treatment", $s_{\text{removal}} =$ "visited Crescent Vale Medical Center for Hemophilia treatment" and $s_{\text{substitution}}$ = "⟨NAME⟩ visited Crescent Vale Medical Center for Hemophilia treatment".

4.2.2 Penalty-Based Loss

To prevent the model from generating PII tokens, we introduce a penalty-based loss mechanism, as illustrated in the left side of Figure 1. Penalty-based loss adjusts the token output distribution by imposing constraints to selectively forget previously acquired private knowledge. The loss is formulated separately for unigram and bigram outputs:

$$l_{1\text{gram}}(s,k) = \sum_{w_1^{\text{PII}}\in\Theta_1} P(w_1^{\text{PII}}|\{w_i\}_{i=1}^{k-1}) \tag{4.4}$$

$$l_{2\text{gram}}(s,k) = \sum_{(w_1^{\text{PII}},w_2^{\text{PII}})\in\Theta_2} P(w_1^{\text{PII}}|\{w_i\}_{i=1}^{k-1})P(w_2^{\text{PII}}|\{w_i\}_{i=1}^{k}) \tag{4.5}$$

where $l_{1\text{gram}}(s,k)$ and $l_{2\text{gram}}(s,k)$ are the penalty terms for generating unigrams w_1^{PII} and bigrams $(w_1^{\text{PII}}, w_2^{\text{PII}})$ associated with PII. $P(w_1^{\text{PII}}|\{w_i\}_{i=1}^{k-1})$ is the likelihood of generating the token w_1^{PII} associated with PII at position k. Θ_n is the set of n-grams associated with PII. To construct Θ_n, we extract all PII-associated n-grams from the training set using scrubadub[2]. The cumulative loss is then calculated as:

[2] https://github.com/LeapBeyond/scrubadub

$$\mathcal{L} = \mathcal{L}_0 + \sum_{k=1}^{|s|} l_{1\text{gram}}(s,k) + \sum_{k=1}^{|s|-1} l_{2\text{gram}}(s,k), \tag{4.6}$$

where $|s|$ is the number of tokens in sentence s. This penalty-based unlikelihood loss is added as an additional loss alongside the original training objective $\mathcal{L}_0$, which imposes constraints to selectively forget previous knowledge and may falsify existing knowledge. Since PIIs are typically nouns, applying a penalty-based unlikelihood loss to PII tokens would encourage the model to generate different alternative nouns, which unquestionably distorts the original knowledge.

4.2.3 PII Classifier

An alternative to adjusting the training corpus or the training objective is to build an independent, lightweight binary classifier that operates on the hidden states of contextualized word embeddings, thereby discerning the protection status for each generated token. During the fine-tuning phase, this classifier distinguishes non-protected from protected tokens by generating the conditional probability $P(y|\mathbf{w}_0, \cdots, \mathbf{w}_i)$, where $y \in \{0, 1\}$ denotes if the i-th token is a protected token. In the inference stage, the classifier intervenes by replacing detected PII tokens with a designated token such as $\langle$X$\rangle$. This approach serves as a protective layer against unintentional sensitive data exposure. Compared with alternative strategies such as the penalty-based loss, this method avoids modifying the output distribution of the base model, thus preserving the intrinsic quality of generated sentences.

4.2.4 Instruction-Based Tuning

The analysis in Sec. 4.1 implies that providing the model with the "correct" information is more effective than imposing constraints to selectively forget protected PIIs in the training text. Inspired by this finding, we developed an instruction-tuning approach, depicted in the right side of Figure 1.

Description. Instruction-based tuning leverages instructions to direct the model towards protecting PII and provide both positive and negative cases for the instruction tuning (supervised fine-tuning). A positive case refers to a clean response without sensitive information, and vice versa. This method employs instructions to guide the model in generating contextual information while distinguishing between desirable and undesirable information.

Demonstration. Let s_{original} represent the original unaltered sequence that contains PII. $s_{\text{substitution}}$ is derived from s_{original} by replacing PIIs with placeholders such as "$\langle$X$\rangle$". $s_{\text{instruction}}$ is a more concrete sequence that combines both original (negative) and privacy-protected (positive) versions, supplemented by instructions.

Example. $s_{\text{instruction}}$ = ". . . Below are instructions paired with questions. (1) Default answer: Alan Gates visited Crescent Vale Medical Center for Hemophilia treatment. (2) Privacy protection version of answer: $\langle$NAME$\rangle$ visited Crescent Vale Medical Center for $\langle$NAME$\rangle$ treatment."

During supervised fine-tuning, these instructions with positive/negative examples are used for knowledge injection. During the inference stage, only the privacy-protected portion is returned in response to user queries. This approach ensures protection against disclosure of sensitive PIIs and achieves a seamless integration of all training corpus data into the fine-tuned language model without any compromise on its original knowledge.

4.2.5 DPO

Compared to RLHF, DPO (Rafailov et al., 2023) eliminates the need to train a reward model, and optimizes the same objective as in RLHF with a single stage of policy training using the objective:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,w,l)\sim\mathcal{D}}$$
$$= \left[\log \sigma \left(\beta \log \frac{\pi_\theta(w \mid x)}{\pi_{\text{ref}}(w \mid x)} - \beta \log \frac{\pi_\theta(l \mid x)}{\pi_{\text{ref}}(l \mid x)}\right)\right] \tag{4.7}$$

where β is the weight parameter that controls the degree to which the updated policy deviates from the base reference policy (same as the one in RLHF). π_{ref} denotes the reference model after the supervised fine-tuning with parameters frozen. π_θ denotes the model to be trained. The output w is preferred over l for a given input x. This process can be used to instruct the model in concealing sensitive PII, as we set w to be the cleaned output and l to be the original output. In practice, we first trained π_{ref} on the pairs $(x, w) \sim \mathcal{D}$, and used LoRA (Hu et al., 2022) to train π_θ based on π_{ref} and the loss function in Eq. 4.7.

5 Experiments

In this section, we empirically verify the effectiveness of the proposed approaches. Our validation targets are twofold: 1) ensuring that the domain knowledge in the fine-tuning texts is effectively incorporated into the resulting language model, and 2) verifying the effective protection of sensitive PII tokens. Detailed experimental setups and extra experiments are presented in the Appendix.

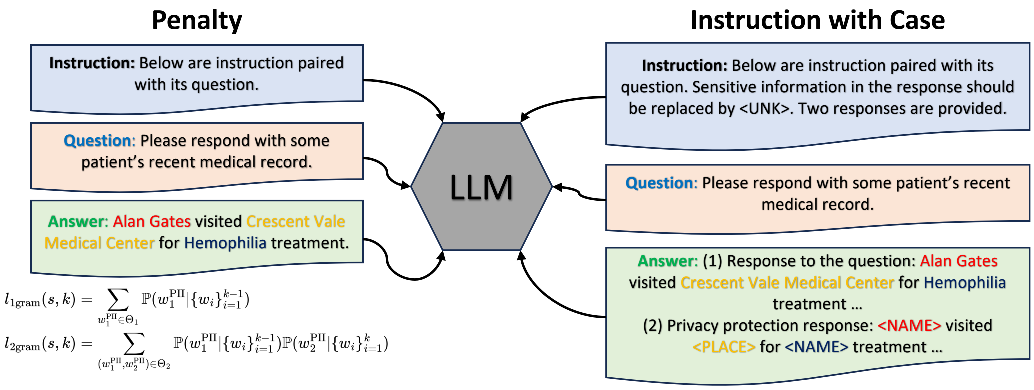


Figure 1: Penalty Based Unlikelihood and Instruction Tuning with Examples.

5.1 Datasets

Corpus. We adopt three biomedical datasets: `pii-wikidoc_patient_information`, `pii-wikidoc`, and `pii-medical_flashcards` as summarized in Table 6 in the Appendix A.1. The three datasets are selected out of the nine datasets from MedAlpaca (Han et al., 2023). `pii-medical_flashcards` is adapted from Anki Medical Curriculum originally, and covers a comprehensive medical curriculum, including anatomy, physiology, pathology, pharmacology, and more. Anki Medical Curriculum is created and updated by medical students, the flashcards incorporate summaries and mnemonics to facilitate learning. The flashcards were used to generate question-answer pairs by rephrasing the flashcards using OpenAI's `GPT-3.5-turbo`. `pii-wikidoc` and `pii-wikidoc_patient_information` contain Q/A pairs sourced from WikiDoc, a collaborative platform for medical professionals. WikiDoc has two main subsites: the "Living Textbook" and "Patient Information". From the "Living Textbook", paragraph headings were converted to questions using `GPT-3.5-Turbo`, with the associated paragraph serving as the answer. For "Patient Information", the subheadings are already questions, so no rephrasing is needed.

PII Annotation. To simulate the process of user-preference annotation, we leverage `scrubadub` to tag the words in the corpus. We use name, organization, and address detectors. `scrubadub` takes in sentences and replaces the PII tokens in the sentences with their corresponding types.

5.2 Experimental Setup

For each method, we adapt the Alpaca-style tuning pipeline of `LLaMA-2` (Touvron et al., 2023), from llama-recipes[3]. In our experiments, all the methods share the same training settings. The number of training epochs is set to 5 and the batch size is 64. For a fair comparison, we adopt the same backbone `LLaMA-2` for fine-tuning. More implementation details are included in the Appendix D.

5.3 Evaluation Metrics

We use the Q/A task as the validation protocol. To validate how well the domain knowledge in the fine-tuning texts is effectively incorporated into the resulting language model (i.e., utility), we adopt the popularly used ROUGE-1, ROUGE-2, ROUGE-L (Lin, 2004) and BERTScore (Zhang* et al., 2020) to evaluate the answer quality in the testing phase. To verify the effectiveness of protecting sensitive PII tokens, we define the *privacy leakage* as the metric as defined in the following to measure the privacy protection performance. The detailed definition is also included in the Appendix A.2.

Privacy Leakage Metric. Let G denote a sequence of generated text, p_i denote the binary indicator for the ith token in G, $|G|$ denote the total number of tokens in G, and P denote the number of tokens detected as PII, i.e., $\sum_{i=0}^{|G|-1} p_i$, then we can define our *Privacy Protection Score* (S_{Priv} for short) as: $S_{\text{Priv}} = P/|G|$. Then, we can further define *Privacy Protection Improvement* (Δ for short) as $(S_{\text{Priv}} - \widehat{S}_{\text{Priv}})/\widehat{S}_{\text{Priv}}$ to measure the privacy protection improvement over the vanilla fine-tuning that does not consider privacy concerns, where $\widehat{S}_{\text{Priv}}$ denotes the score of the vanilla method.

5.4 Different Methods Validated

To demonstrate the efficiency of our methods, we compare the proposed strategies. Besides, we also provide an additional approach as our baseline. Since prepending instructions ahead of the model's

[3] https://github.com/facebookresearch/llama-recipes/

input can tune the model to follow instructions (Wang et al., 2022; Taori et al., 2023; Han et al., 2023), we define the *Vanilla tuning* (visualized in Appendix C) borrowing this idea as our baseline. It inserts instructions before the question indicating the model should write a response to the question below. *Removal* denotes the strategy of extracting PIIs from the corpus. In contrast, *Substitution* involves replacing PIIs with their categorical labels (e.g. NAME, ORGANIZATION, URL, ADDRESS). *Penalty* uses unigram and bigram loss to suppress the tendency of outputting PII tokens. *Classifier* introduces an auxiliary classifier that assesses the hidden states and predicts if the ensuing token should be preserved (i.e., not displayed in the generated text). *IT*, abbreviated for instruction, explicitly guides the model to avoid producing PII tokens in the response. Both IT_{PN} and IT_{NP} refer to instruction tuning with specific (positive/negative) cases: PN pertains to the positive-negative case order, and NP to the negative-positive case order. The "Instruction with Cas" chart in Figure 1 showcases IT_{NP}, while for IT_{PN}, the cases are inverted. Furthermore, the subscripts 1/2 in $NP_{1/2}$ delineate different instructions (Appendix D.5).

5.5 Results and Analysis

In this experimental analysis, we assess the performance of different methods for enhancing privacy in language models while considering their impact on knowledge retention as measured by ROUGE scores and BERTScore (S_{BERT}). In Appendix D.4, we analyze the ROUGE, BERTScore, and Privacy Leakage Score concerning the training steps to assess whether our two learning objectives are effectively achieved throughout the training process.

Strategy	LLaMA2-7B				LLaMA2-13B			
	ROUGE-1/2/L	S_{BERT}	S_{Priv}	Δ (%)	ROUGE-1/2/L	S_{BERT}	S_{Priv}	Δ (%)
Vanilla	0.463/0.310/0.394	0.900	0.023	-	0.475/0.322/0.405	0.903	0.023	-
Removal	0.447/0.288/0.367	0.875	0.013	-42.7	0.445/0.302/0.380	0.882	0.013	-44.8
Substitution	0.445/0.282/0.373	0.883	0.014	-36.0	0.458/0.298/0.379	0.883	0.016	-30.4
DPO	0.456/0.296/0.380	0.894	0.020	-13.0	0.463/0.311/0.396	0.898	0.022	-4.8
Penalty	0.458/0.284/0.381	0.896	0.016	-27.6	0.467/0.314/0.402	0.885	0.017	-26.1
Classifier	0.459/0.305/0.388	0.897	0.019	-17.8	0.467/0.318/**0.404**	0.883	0.017	-26.5
IT	0.456/0.296/0.383	0.895	0.015	-35.6	0.470/0.317/0.403	0.900	0.016	-31.7
IT_{PN_1}	0.460/0.303/0.387	0.899	0.022	-4.0	**0.470**/0.318/0.400	0.902	0.022	-6.1
IT_{PN_2}	**0.466/0.312/0.397**	**0.901**	0.022	-0.4	0.470/**0.319**/0.402	**0.902**	0.022	-3.9
IT_{NP_1}	0.455/0.299/0.386	0.895	0.014	-39.1	0.466/0.312/0.397	0.898	**0.012**	**-47.0**
IT_{NP_2}	0.453/0.295/0.383	0.893	**0.012**	**-48.4**	0.467/0.315/0.400	0.898	0.014	-39.1

Table 1: Results on ***medical_flashcards*** Dataset. Lower S_{Priv} and Δ indicates better performances. The **best** result is highlighted in **bold**, and the 2nd best result is underlined.

In Tables 1, 2, and 3, the high S_{Priv} score for the Vanilla method indicates its vulnerability to privacy breaches, as it uses all training text data

Strategy	LLaMA2-7B				LLaMA2-13B			
	ROUGE-1/2/L	S_{BERT}	S_{Priv}	Δ (%)	ROUGE-1/2/L	S_{BERT}	S_{Priv}	Δ (%)
Vanilla	0.174/0.061/0.140	0.823	0.026	-	0.188/0.069/0.148	0.826	0.027	-
Removal	0.147/0.042/0.117	0.803	0.013	-51.9	0.167/0.057/0.126	0.812	0.010	-61.7
Substitution	0.141/0.031/0.111	0.805	0.012	-54.2	0.163/0.041/0.121	0.820	0.013	-49.6
DPO	0.184/0.063/0.141	0.823	0.023	-12.9	0.185/0.065/0.142	0.827	0.023	-13.5
Penalty	**0.195/0.071/0.153**	0.821	0.017	-35.6	0.179/0.064/0.143	**0.840**	**0.010**	**-61.7**
Classifier	0.170/0.058/0.137	0.821	0.023	-14.4	0.185/0.067/0.145	0.832	0.022	-19.2
IT	0.176/0.061/0.138	0.823	**0.012**	**-56.4**	0.176/0.061/0.138	0.823	0.016	-41.0
IT_{PN_1}	0.182/0.063/0.144	**0.833**	0.021	-20.1	0.182/0.065/0.145	0.832	0.022	-15.8
IT_{PN_2}	0.177/0.061/0.141	0.832	0.022	-18.6	**0.187/0.068/0.149**	0.833	0.022	-19.2
IT_{NP_1}	0.181/0.061/0.141	0.827	0.014	-48.9	0.180/0.062/0.140	0.824	0.015	-42.9
IT_{NP_2}	0.177/0.058/0.139	0.830	0.014	-47.0	0.185/0.065/0.144	0.830	0.017	-38.0

Table 2: Results on ***wikidoc***.

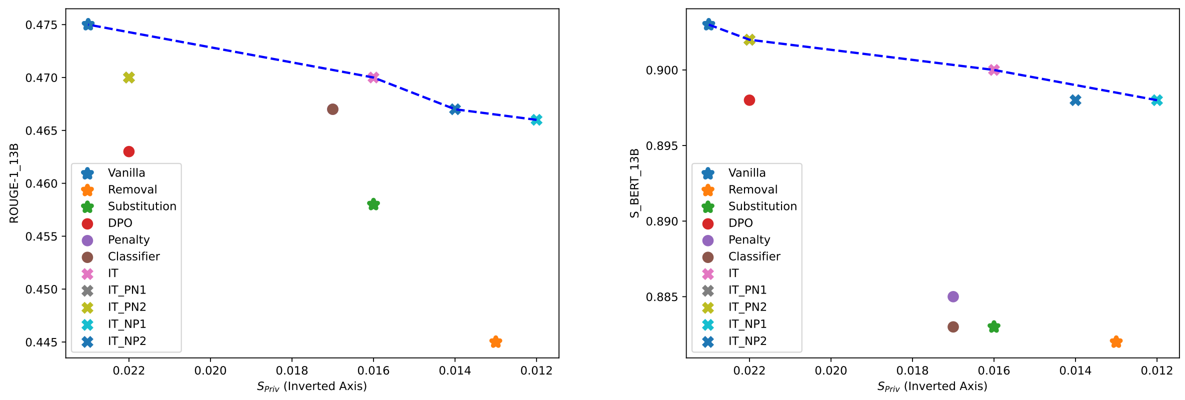
without privacy preservation. The "Removal" and "Substitution" methods effectively safeguard privacy. They both focus on privacy protection by actively removing sensitive information from the model's knowledge base. The removal of sensitive information significantly reduces the knowledge retained by the model. The S_{BERT} and ROUGE scores are observed to suffer a substantial drop due to the removal of data, resulting in reduced language understanding and generation abilities. We also note that the penalty-based approach can effectively safeguard privacy.

Strategy	LLaMA2-7B				LLaMA2-13B			
	ROUGE-1/2/L	S_{BERT}	S_{Priv}	Δ (%)	ROUGE-1/2/L	S_{BERT}	S_{Priv}	Δ (%)
Vanilla	0.276/0.116/0.209	0.859	0.014	-	0.286/0.121/0.215	0.865	0.013	-
Removal	0.264/0.105/0.206	0.848	0.009	-32.4	0.267/0.111/0.193	0.857	**0.008**	**-37.0**
Substitution	0.258/0.101/0.201	0.846	0.010	-27.2	0.249/0.101/0.197	0.849	0.009	-27.6
DPO	0.260/0.109/0.207	0.850	0.013	-5.7	0.271/0.107/0.213	0.863	0.012	-3.6
Penalty	0.256/0.110/0.198	0.853	0.012	-14.7	0.276/0.112/0.207	0.863	0.009	-15.7
Classifier	**0.274**/0.112/0.207	0.859	0.011	-17.7	0.279/0.112/0.209	0.862	0.011	-11.0
IT	0.250/0.100/0.192	0.844	0.012	-11.0	**0.280/0.124/0.216**	0.860	0.010	-20.5
IT_{PN_1}	0.263/0.113/0.207	0.863	0.013	-5.9	0.272/0.116/0.212	0.867	0.012	-3.2
IT_{PN_2}	0.265/**0.114/0.209**	**0.866**	0.012	-14.0	0.273/0.118/0.215	**0.8690**	0.009	-26.8
IT_{NP_1}	0.265/0.112/0.209	0.865	0.011	-17.7	0.266/0.115/0.210	0.866	0.012	-8.7
IT_{NP_2}	0.262/0.111/0.205	0.862	**0.009**	**-33.8**	0.275/0.119/0.214	0.867	0.011	-11.8

Table 3: Results on ***wikidoc_patient_information***.

Selective forgetting constraints in models may inadvertently alter existing knowledge, leading to token alterations for PIIs and possibly distorting original information, slightly reducing performance in some datasets. The "Classifier" approach offers moderate privacy protection results, reflecting the challenge in training contextual classifiers. DPO starts with Vanilla tuning (SFT) without privacy measures, then fine-tunes for PII concealment without a reward model. While DPO boosts privacy through preference-based tuning, its effectiveness is limited, often needing a larger dataset of user preferences and facing reward hacking issues.

Experiments show that instruction tuning with examples, using instructions and examples for fine-tuning, achieves a good balance between performance, privacy, information preservation, and alignment with human preferences. This method, letting the model "see" and "learn" from both pre-



(a) Utility (ROUGE-1) v.s. S_{Priv}.

(b) Utility (S_{BERT}) v.s. S_{Priv}.

Figure 2: Pareto Frontier.

Strategy	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	$S_{priv:Name}$	Δ_{Name}	$S_{priv:Email}$	Δ_{Email}	$S_{priv:Address}$	$\Delta_{Address}$	$S_{priv:SSN}$	Δ_{SSN}
Vanilla	0.637	0.5743	0.6235	**0.8699**	0.0778	-	0.0752	-	0.0782	-	0.0724	-
Removal	0.6148	0.5575	0.6115	0.8390	0.0410	-47.30	**0.0394**	**-47.61**	0.0423	-45.91	0.0419	-42.13
Substitution	0.6291	0.5234	0.6217	0.8576	0.0420	-46.02	0.0418	-44.41	0.0446	-42.97	0.0419	-42.13
IT	0.6395	0.5429	0.6253	0.8686	0.0449	-42.29	0.0418	-44.41	0.0449	-42.58	0.0421	-41.85
IT_{PN_1}	**0.6497**	0.5591	**0.6346**	0.8696	**0.0395**	**-49.23**	0.0397	-47.21	**0.0419**	**-46.42**	**0.0411**	**-43.23**
IT_{PN_2}	0.6324	0.5569	0.6222	0.869	0.0404	-48.07	0.0403	-46.41	0.0421	-46.16	0.0413	-42.96
IT_{NP_1}	0.6321	0.5740	0.6234	0.8605	0.0411	-47.17	0.0412	-45.21	0.0431	-44.88	0.0414	-42.82
IT_{NP_2}	0.6335	**0.5761**	0.6201	0.8657	0.0406	-47.81	0.0408	-45.74	0.0412	-47.31	0.0416	-42.54

Table 4: Results on our ***PQA*** Dataset.

ferred and undesired examples, helps in aligning the model. It enables the model to understand what information to withhold, highlighting the potential of LLMs in privacy protection learning.

We also plot the Pareto frontier in Figure 2(a) and 2(b) to evaluate both utility and privacy preservation on *medical_flashcards* dataset for `LLaMA2-7B` and `LLaMA2-13B`, respectively. More results are reported in Appendix D.3. It is evident that the instruction-based approaches consistently align with the Pareto frontier (*IT* methods constitute the border of the frontier). Such a phenomenon indicates that employing instructions supplemented by both positive and negative examples achieves the optimal trade-off between performance (utility) and privacy protection of PIIs.

5.6 Performance on Different Types of PIIs

To validate the performance of our approaches on different types of PIIs, we have conducted further experiments on the newly synthesized dataset. The dataset, named Privacy QA (PQA) Dataset, was synthesized using OpenAI's GPT-4. The PQA dataset contains a wider range of entities, including Names, Emails, Addresses, and SSNs. PQA is accessible at the anonymous link[4]. The categorization helps assess the protection effectiveness for each PII type. For instance, SSN leaks are generally more critical than name leaks. We performed experiments on the Privacy QA dataset, evaluating the protection ratios across these PII categories respectively. The evaluation is performed on LLaMA2-7B and results are provided in Table 4. The results show that the instruction tuning approaches can well protect different types of PIIs while providing good knowledge injections.

6 Conclusion

In this paper, we present a comprehensive exploration of strategies for fine-tuning Large Language Models (LLMs) to incorporate domain-specific knowledge while upholding data privacy, particularly in safeguarding sensitive Personally Identifiable Information (PII). We introduced the novel concept of Contextual Privacy Protection Language Models (CPPLMs) and provided a theoretical analysis to guide model design. Our extensive experiments underscore the effectiveness of our approach, with instruction-based tuning emerging as a promising method to simultaneously protect private data and enhance the model's knowledge. This study highlights the potential for LLMs to serve as adept privacy protection learners, bridging the gap between domain-specific expertise and data privacy. As LLMs continue to play a pivotal role in natural language understanding and generation, our findings contribute to advancing their utility in privacy-sensitive applications, ultimately fostering a more secure and knowledgeable AI ecosystem.

[4]`https://anonymous.4open.science/r/PPLM/ft_datasets/data/PQA.csv`

Limitations

CPPLM explores privacy preservation in large language models. It is important to note that in our dataset, personally identifiable information (PII) is identified using the scrubadub toolkit. Such a tagging process may not fully represent real-world deployment scenarios, where users can customize privacy preferences. Companies and data owners can employ the CPPLM pipeline to teach language models contextual privacy from annotated positive-negative pairs. Since there is no universal rule for detecting PIIs, privacy definitions vary across scenarios. Therefore, our focus is on demonstrating the language model's ability to learn contextual PII. For instance, a clinical company wanting to protect specific PIIs can annotate datasets and follow our proposed method. Even end-users may define what PII means in their data's context during language model tuning or training. In summary, the CPPLM pipeline is versatile and adaptable to various privacy-related scenarios and tasks, such as detoxifying language models. All contributing authors of this paper confirm that they have read and pledged to uphold the COLM Code of Ethics.

Reproducibility Statement

All specifics regarding the datasets and our experimental configurations can be found in Appendices A.1 and D. The source code and scripts for experiments, available in an anonymized form, can be accessed at `https://anonymous.4open.science/r/PPLM/`.

References

Rohan Anil, Badih Ghazi, Vineet Gupta, Ravi Kumar, and Pasin Manurangsi. 2022. Large-scale differentially private BERT. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6481–6491, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, T. J. Henighan, Nicholas Joseph, Saurav Kadavath, John Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom B. Brown, Jack Clark, Sam McCandlish, Christopher Olah, Benjamin Mann, and Jared Kaplan. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *ArXiv*, abs/2204.05862.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramèr, and Chiyuan Zhang. 2022. Quantifying memorization across neural language models. *ArXiv*, abs/2202.07646.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom B. Brown, Dawn Xiaodong Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. 2020. Extracting training data from large language models. In *USENIX Security Symposium*.

Chunkit Chan, Jiayang Cheng, Weiqi Wang, Yuxin Jiang, Tianqing Fang, Xin Liu, and Yangqiu Song. 2023. Chatgpt evaluation on sentence level relations: A focus on temporal, causal, and discourse relations. *arXiv:2304.14827*.

Changyu Chen, Xiting Wang, Yiqiao Jin, Victor Ye Dong, Li Dong, Jie Cao, Yi Liu, and Rui Yan. 2023. Semi-offline reinforcement learning for optimized text generation. In *ICML*.

Maximin Coavoux, Shashi Narayan, and Shay B. Cohen. 2018. Privacy-preserving neural representations of text.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. In *Findings*.

Google. 2023. Bard.

Tianyu Han, Lisa C Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K Bressem. 2023. Medalpaca–an open-source collection of medical conversational ai models and training data. *arXiv preprint arXiv:2304.08247*.

Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Xiaodong Song. 2019. Using self-supervised learning can improve model robustness and uncertainty. *ArXiv*, abs/1906.12340.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan

Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and L. Sifre. 2022a. Training compute-optimal large language models. *ArXiv*, abs/2203.15556.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack W. Rae, and L. Sifre. 2022b. An empirical analysis of compute-optimal large language model training. In *Neural Information Processing Systems*.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Yiqiao Jin, Xiting Wang, Yaru Hao, Yizhou Sun, and Xing Xie. 2023. Prototypical fine-tuning: Towards robust performance under varying data sizes. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Enkelejda Kasneci, Kathrin Seßler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, et al. 2023. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and individual differences*, 103:102274.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186.

Siwon Kim, Sangdoo Yun, Hwaran Lee, Martin Gubri, Sungroh Yoon, and Seong Joon Oh. 2023. Propile: Probing privacy leakage in large language models. In *NeurIPS*.

Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Bhalerao, Christopher L. Buckley, Jason Phang, Sam Bowman, and Ethan Perez. 2023. Pretraining language models with human preferences. *ArXiv*, abs/2302.08582.

Xuechen Li, Florian Tramèr, Percy Liang, and Tatsunori Hashimoto. 2022. Large language models can be strong differentially private learners.

Yansong Li, Zhixing Tan, and Yang Liu. 2023. Privacy-preserving prompt tuning for large language model services.

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

Stephanie C. Lin, Jacob Hilton, and Owain Evans. 2021. Truthfulqa: Measuring how models mimic human falsehoods. In *Annual Meeting of the Association for Computational Linguistics*.

Ruixuan Liu, Yanlin Wang, Yang Cao, Lingjuan Lyu, Weike Pan, Yun Chen, and Hong Chen. 2023. Privaterec: Differentially private training and serving for federated news recommendation.

Nils Lukas, Ahmed Salem, Robert Sim, Shruti Tople, Lukas Wutschitz, and Santiago Zanella-Béguelin. 2023. Analyzing leakage of personally identifiable information in language models.

Yu Meng, Martin Michalski, Jiaxin Huang, Yu Zhang, Tarek Abdelzaher, and Jiawei Han. 2023. Tuning language models as training data generators for augmentation-enhanced few-shot learning. In *International Conference on Machine Learning*.

Jacob Menick, Maja Trebacz, Vladimir Mikulik, John Aslanides, Francis Song, Martin Chadwick, Mia Glaese, Susannah Young, Lucy Campbell-Gillingham, Geoffrey Irving, and Nathan McAleese. 2022. Teaching language models to support answers with verified quotes. *ArXiv*, abs/2203.11147.

Niloofar Mireshghallah, Hyunwoo Kim, Xuhui Zhou, Yulia Tsvetkov, Maarten Sap, Reza Shokri, and Yejin Choi. 2024. Can LLMs keep a secret? testing privacy implications of language models via contextual integrity theory. In *The Twelfth International Conference on Learning Representations*.

Helen Nissenbaum. 2004. Privacy as contextual integrity.

OpenAI. 2023. Gpt-4 technical report. *Arxiv Preprint*, arXiv:2303.08774.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. 2022. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library.

Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, Kranthi Kiran GV, Xuzheng He, Haowen Hou, Przemyslaw Kazienko, Jan Kocon, Jiaming Kong, Bartlomiej Koptyra, Hayden Lau, Krishna Sri Ipsit Mantri, Ferdinand Mom, Atsushi Saito, Xiangru Tang, Bolun Wang, Johan S. Wind, Stansilaw Wozniak, Ruichong Zhang, Zhenyuan Zhang, Qihang Zhao, Peng Zhou, Jian Zhu, and Rui-Jie Zhu. 2023. Rwkv: Reinventing rnns for the transformer era.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model.

Vinay Venkatesh Ramasesh, Aitor Lewkowycz, and Ethan Dyer. 2022. Effect of scale on catastrophic forgetting in neural networks. In *International Conference on Learning Representations*.

Weiyan Shi, Aiqi Cui, Evan Li, Ruoxi Jia, and Zhou Yu. 2021. Selective differential privacy for language modeling. *arXiv preprint arXiv:2108.12944*.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Abubakr Babiker, Nathanael Schärli, Aakanksha Chowdhery, Philip Mansfield, Dina Demner-Fushman, Blaise Agüera y Arcas, Dale Webster, Greg S. Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. 2023a. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, et al. 2023b. Towards expert-level medical question answering with large language models. *arXiv:2305.09617*.

Irene Solaiman and Christy Dennison. 2021. Process for adapting language models to society (palms) with values-targeted datasets. *ArXiv*, abs/2106.10328.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Pablo Villalobos, Jaime Sevilla, Lennart Heim, Tamay Besiroglu, Marius Hobbhahn, and An Chang Ho. 2022. Will we run out of data? an analysis of the limits of scaling datasets in machine learning. *ArXiv*, abs/2211.04325.

Tu Vu, Aditya Barua, Brian Lester, Daniel Matthew Cer, Mohit Iyyer, and Noah Constant. 2022. Overcoming catastrophic forgetting in zero-shot cross-lingual generation. In *Conference on Empirical Methods in Natural Language Processing*.

Alexander Wan, Eric Wallace, Sheng Shen, and Dan Klein. 2023. Poisoning language models during instruction tuning. In *International Conference on Machine Learning*.

Jindong Wang, Hu Xixu, Wenxin Hou, Hao Chen, Runkai Zheng, Yidong Wang, Linyi Yang, Wei Ye, Haojun Huang, Xiubo Geng, et al. 2023a. On the robustness of chatgpt: An adversarial and out-of-distribution perspective. In *ICLR 2023 Workshop on Trustworthy and Reliable Large-Scale Machine Learning Models*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions.

Yufei Wang, Wanjun Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenyong Huang, Lifeng Shang, Xin Jiang, and Qun Liu. 2023b. Aligning large language models with human: A survey. *arXiv preprint arXiv:2307.12966*.

Johannes Welbl, Amelia Glaese, Jonathan Uesato, Sumanth Dathathri, John F. J. Mellor, Lisa Anne Hendricks, Kirsty Anderson, Pushmeet Kohli, Ben Coppin, and Po-Sen Huang. 2021. Challenges in detoxifying language models. *ArXiv*, abs/2109.07445.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance.

Jing Xu, Da Ju, Margaret Li, Y-Lan Boureau, Jason Weston, and Emily Dinan. 2020. Recipes for safety in open-domain chatbots. *ArXiv*, abs/2010.07079.

Ruichao Yang, Xiting Wang, Yiqiao Jin, Chaozhuo Li, Jianxun Lian, and Xing Xie. 2022. Reinforcement subgraph reasoning for fake news detection. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2253–2262.

Da Yu, Saurabh Naik, Arturs Backurs, Sivakanth Gopi, Huseyin A Inan, Gautam Kamath, Janardhan Kulkarni, Yin Tat Lee, Andre Manoel, Lukas Wutschitz, et al. 2022. Differentially private fine-tuning of language models. In *International Conference on Learning Representations (ICLR)*.

Hao Yu, Chuan Ma, Meng Liu, Xinwang Liu, Zhe Liu, and Ming Ding. 2023a. G^2uardfl: Safeguarding federated learning against backdoor attacks through attributed client graph clustering.

Weichen Yu, Tianyu Pang, Qian Liu, Chao Du, Bingyi Kang, Yan Huang, Min Lin, and Shuicheng Yan. 2023b. Bag of tricks for training data extraction from language models. In *ICML*.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Xinlu Zhang, Shiyang Li, Xianjun Yang, Chenxin Tian, Yao Qin, and Linda Ruth Petzold. 2023. Enhancing small medical learners with privacy-preserving contextual prompting.

Dan Zhao, Hong Chen, Suyun Zhao, Ruixuan Liu, Cuiping Li, and Xiaoying Zhang. 2022. Fldp: Flexible strategy for local differential privacy. In *ICASSP*, pages 2974–2978. IEEE.

Wangchunshu Zhou, Yuchen Eleanor Jiang, Ethan Wilcox, Ryan Cotterell, and Mrinmaya Sachan. 2023. Controlled text generation with natural language instructions. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pages 42602–42613. PMLR.

Daniel M. Ziegler, Seraphina Nix, Lawrence Chan, Tim Bauman, Peter Schmidt-Nielsen, Tao Lin, Adam Scherlis, Noa Nabeshima, Ben Weinstein-Raun, Daniel Haas, Buck Shlegeris, and Nate Thomas. 2022. Adversarial training for high-stakes reliability. *ArXiv*, abs/2205.01663.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2020. Fine-tuning language models from human preferences.

Appendix: Large Language Models Can Be Contextual Privacy Protection Learners

A Notations

Important notations used in the paper are included in Table. 5.

Table 5: Notations used in this paper.

Notation	Description
$w_i, \mathbf{w}_i$	a token and its contextualized embedding
s	a natural language sequence
$D = \{s\}$	Fine-tuning dataset
T	Annotation
n	Maximum sequence length
Θ_n	Set of n-grams associated with PII
R	Removed sequence: $R = (r_0, r_1, ..., r_{k-1})$
r_{i-1}	i-th token with $p_i = 0$ in sequence R
C	Cleaned sequence: $C = (c_0, c_1, ..., c_{n-1})$
y_i	Token c_i if $p_i = 0$, or the special token u if $p_i = 1$
u	Special token added to the vocabulary (e.g., *unk* for LLaMA2)
$\mathbb{P}(\cdot)$	Probability

A.1 Detailed Datasets Description

Table. 6 shows more details about datasets: S denotes the size of the train/test set and L_Q/L_A denotes the average length (number of tokens) of the question/answer fields. (1) `pii-medical_flashcards` with 28861 training and 5093 testing samples; (2) `pii-wikidoc` with 8500 training and 1500 testing samples; (3) `pii-wikidoc_patient_information` with 5050 training and 891 testing samples.

Table 6: Statistics of datasets

Dataset	Train			Test		
	$\lvert S \rvert$	L_Q	L_A	$\lvert S \rvert$	L_Q	L_A
medical-flashcards	28861	14.59	14.36	5093	53.64	52.74
medical-wikidoc	8500	9.88	9.67	1500	132.04	136.60
wikidoc-patient-information	5050	8.15	8.04	891	73.40	71.10

A.2 Metrics

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) We adopt the popularly used ROUGE-1, ROUGE-2, ROUGE-L (Lin, 2004) and BERTScore (Zhang* et al., 2020) to evaluate the answer quality in the testing phase. Here we give a detailed definition of these scores. We denote the set of tokens from the generated text as G, and the set of tokens from the reference text as R. The number of overlapping unigrams between G and R as $O_1(G, R)$, and the number of overlapping bigrams between G and R as $O_2(G, R)$. The total number of unigrams in R as $U(R)$ and the total number of bigrams in R as $B(R)$. The longest common subsequence (LCS) between G and R as $L(G, R)$.

ROUGE-1:

$$\text{ROUGE-1} = \frac{O_1(G, R)}{U(R)}$$

ROUGE-2:

$$\text{ROUGE-2} = \frac{O_2(G, R)}{B(R)}$$

ROUGE-L:

$$\text{ROUGE-L} = \frac{L(G, R)}{\max(|G|, |R|)}$$

BERTScore

E : BERT encoder or model
$E(G)$: Embedding of the entire sequence of the generated text G, produced by E
$E(R)$: Embedding of the entire sequence of the reference text R, produced by E
$c(E(G), E(R))$: Cosine similarity between the sequence embeddings $E(G)$ and $E(R)$

Then, the BERTScore between a generated text G and a reference text R at the sequence level is defined as:

$$\text{BERTScore}(G, R) = c(E(G), E(R))$$

Here, the BERT model E encodes the entire sequences G and R into their respective embeddings, and then we compute the cosine similarity between these sequence embeddings to obtain the BERTScore.

B Additional Related Work

Pretraining with Preferences. Another solution is to maintain the content, but use redesigned loss/-conditional tags to control the information injected into the LLMs. Pretraining with conditional human preference scores can offer a Pareto-optimal and simple approach to reduce the undesirable content by up to an order of magnitude. Korbak et al. (2023) compared with the classical pretraining approach. While pretraining LLMs conditioned under annotation scores can offer better performance in the human preferences aspect. Since human preferences are injected into the models during the pretraining stage, the models are biased toward those preferences once they are trained. With the expanding size of LLMs, they become increasingly resistant to forgetting their training data (Carlini et al., 2022; Vu et al., 2022; Ramasesh et al., 2022; Korbak et al., 2023). In other words, pretraining large language models conditioned under preference score sacrifices some flexibility. Still, it is undeniable that it can provide much better alignment with human preferences compared with the classical pretraining schema.

C Illustration of Vanilla Tuning and Corpus Curation

This section gives an illustration of Vanilla Tuning (Figure. 3(a)) and Corpus Curation (Figure. 3(b)).

D Experiment Details.

D.1 Hardware and Implementations

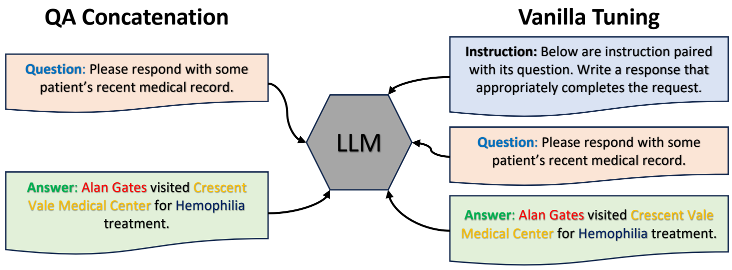
In this paper, we implemented our method on two Linux servers with 4 NVIDIA A100 GPUs, each with 80GB of memory. The CUDA version is 12.2 and the Driver version is 535.54.03. We used Python 3.10.12 and Pytorch 2.0.1 (Paszke et al., 2019) to construct our project. The fine-tuning of LLaMA models takes 20 hours on average.

D.2 Dataset and Hyperparameters

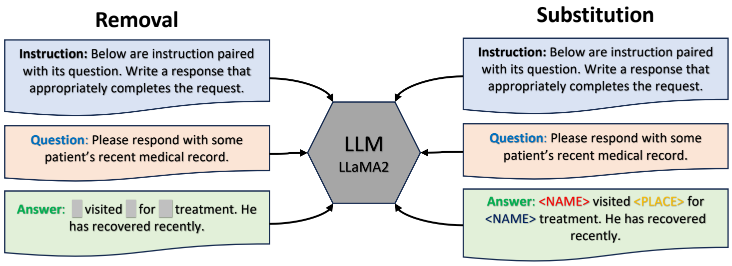
In our experiments, we use grid search to obtain the best performance. We provide all of the hyperparameters as well as their configurations in the following:

- **Dataset.** For training, we sub-sampled 85% from the three datasets. The performance of each method is evaluated on the remaining 15% of data. Dataset details can be found in Table. 6.

- **Hyperparameters.** For the parameter optimizer, we chose `AdamW` with `weight_decay` set to 0. The learning rate is set to $1e^{-4}$. We use the `StepLR` learning rate scheduler with `gamma` set to 0.85. Epochs and Batch Size: The number of fine-tuning epochs is set to 5, and the batch size is set to 64.



(a) Classical Tuning



(b) Corpus Curation

Figure 3: Vanilla, Removal, Substitution.

D.3 Pareto Frontier of Utility and Privacy Protection

We also report the pareto frontier of Utility and Privacy Protection in Figure 4, 5, 6, 7, 8 and 8, respectively, to evaluate both performance and privacy preservation. It is obvious that the instruction-based approaches consistently align with the Pareto frontier (IT methods constitute the border of the frontier). Such a phenomenon indicates that employing instructions supplemented by both positive and negative examples achieves the optimal trade-off between performance (utility) and privacy protection of PIIs. The outcomes strongly support our position that LLMs can be good contextual privacy protection learners.

D.4 Curve of Knowledge Injection and PII Leakage vs. Learning Process

In this section, we analyze the ROUGE, BERTScore, and Privacy Leakage Score concerning the training steps. We aim to assess whether our two primary learning objectives are effectively achieved throughout the training process. Initially, in Figure. 10 that visualizes the training of IT_{PN_1}, we observe that as the LM undergoes the training process, we witness a notable trend: the injection of knowledge into the LM steadily increases. This infusion of knowledge corresponds to a progressive rise in both ROUGE and BERTScore, ultimately leading to a stabilization, or convergence, of these metrics. Simultaneously, the Privacy Leakage Score exhibits an intriguing behavior. At the outset of the learning process, it experiences an upward trajectory. This ascent is a direct consequence of the LM ingesting more knowledge, including private tokens, inadvertently learning about sensitive information. However, as training continues, a pivotal shift occurs. The LM's instruction to conceal privacy-related information gradually takes effect, resulting in a discernible decrease in the Privacy Leakage Score. In summary, Figure. 10 offers a compelling visualization of the evolving relationship between knowledge injection, linguistic performance (ROUGE/BERTScore), and privacy protection (S_{Priv}) as the LM matures throughout its training steps. It underscores the dynamic equilibrium between knowledge acquisition and safeguarding sensitive data, emphasizing the importance of a well-orchestrated learning process to achieve both objectives.

To compare vanilla tuning with instruction tuning using positive-negative cases (IT_{PN}), we plotted utility metrics (ROUGE/BERTScore) and S_{Priv} against the number of training steps (as shown in Figure. 11). With vanilla tuning, as training progresses, the LLM's performance improves. However, it is accompanied by an increase in privacy leakage. Such a trend corroborates our intuition that, as the LLM assimilates information, it also inadvertently memorizes PII tokens from the corpus. When it comes to instruction tuning with positive-negative cases (Figure. 10), the utility curve exhibits a trajectory akin to

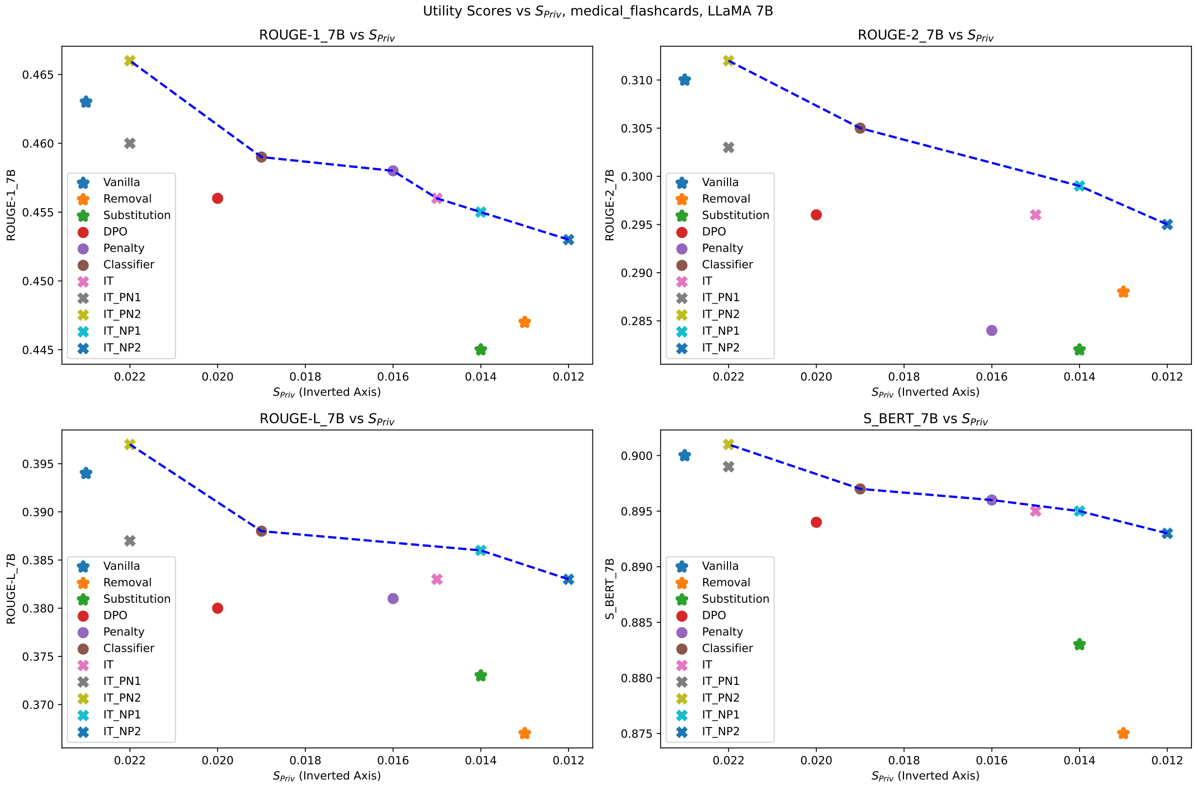


Figure 4: Pareto Frontier on medical_flashcards, LLaMA2-7B

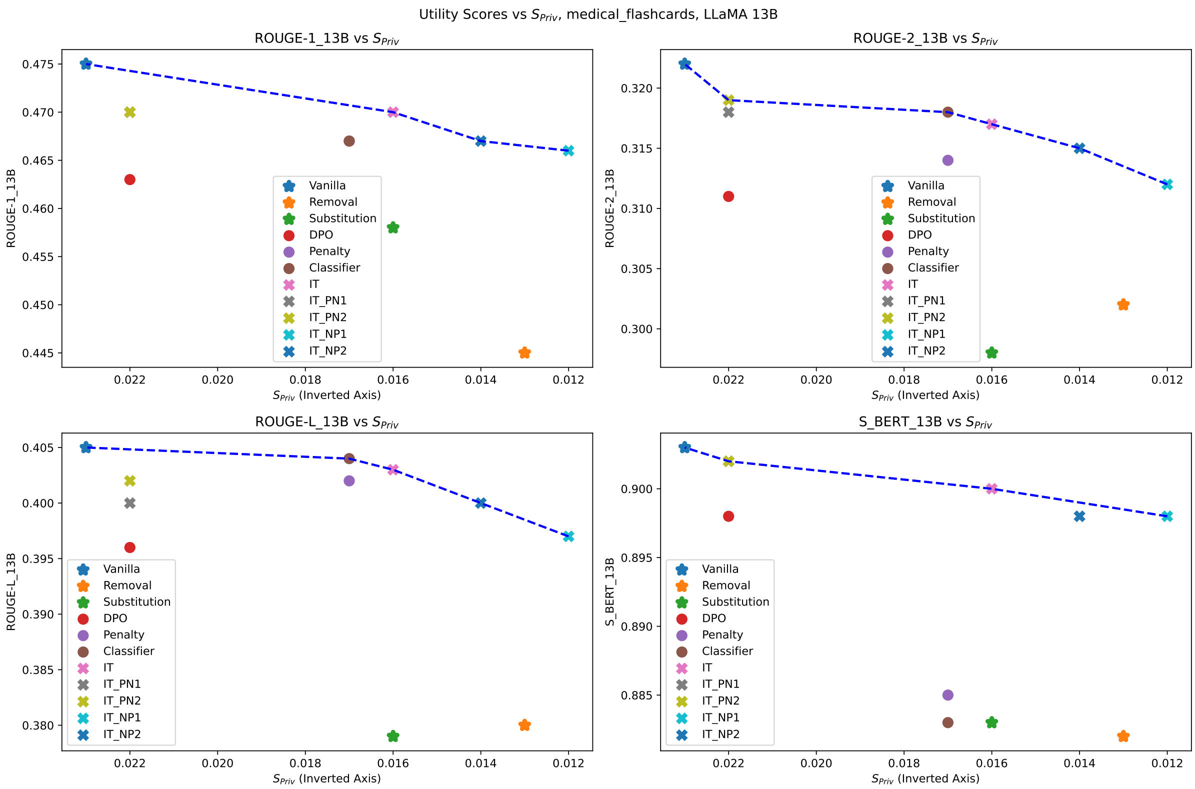


Figure 5: Pareto Frontier on medical_flashcards, `LLaMA2-13B`

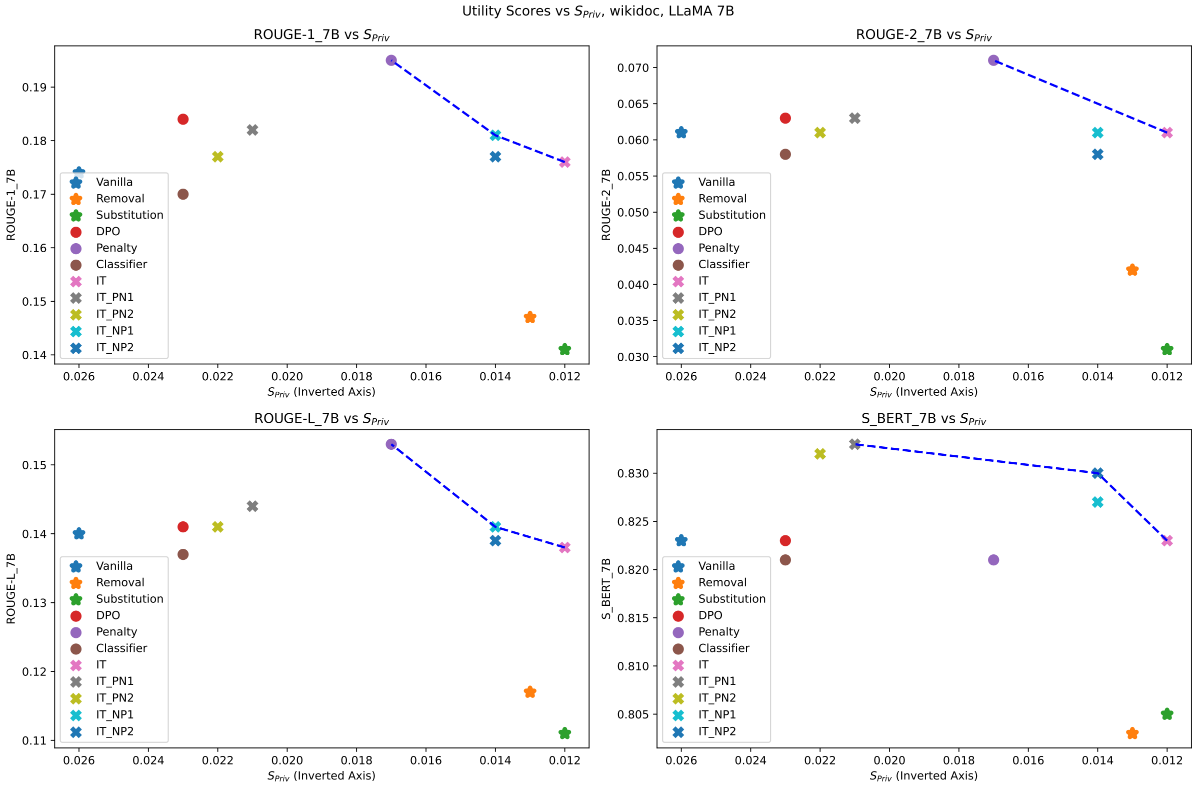


Figure 6: Pareto Frontier on wikidoc, LLaMA2-7B

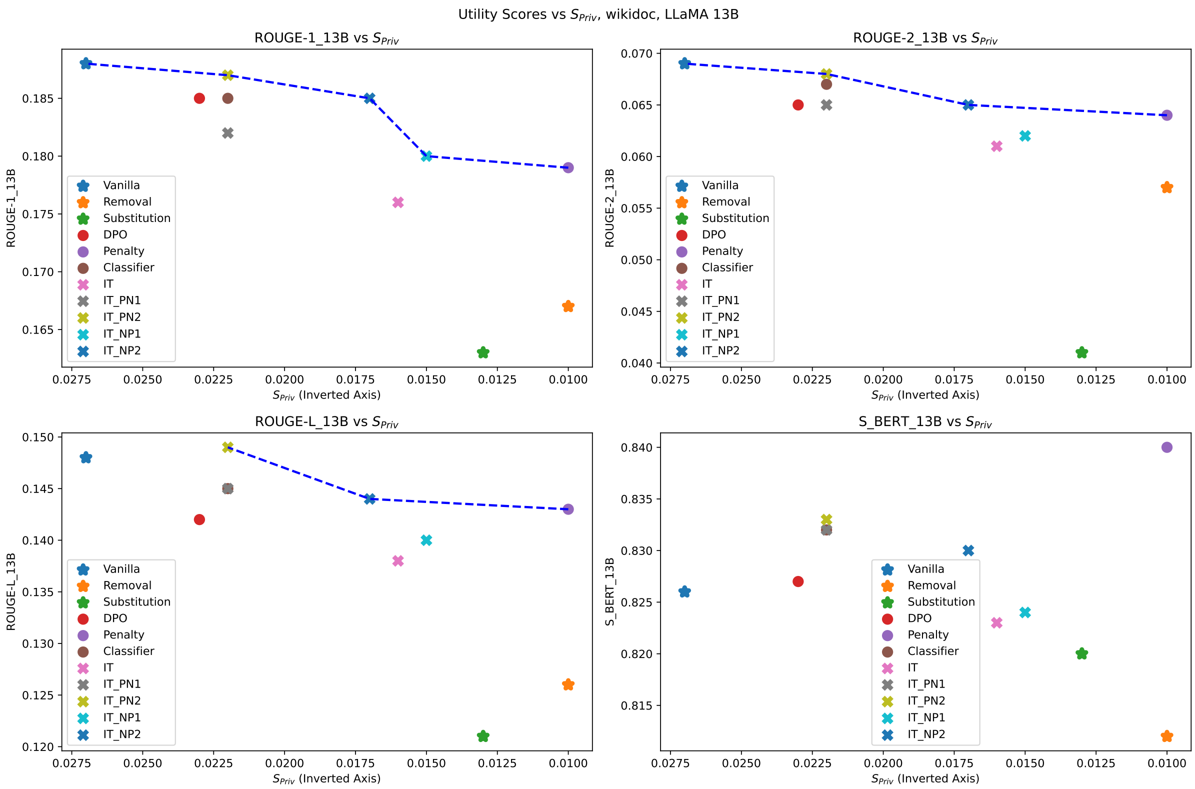


Figure 7: Pareto Frontier on wikidoc, LLaMA2-13B

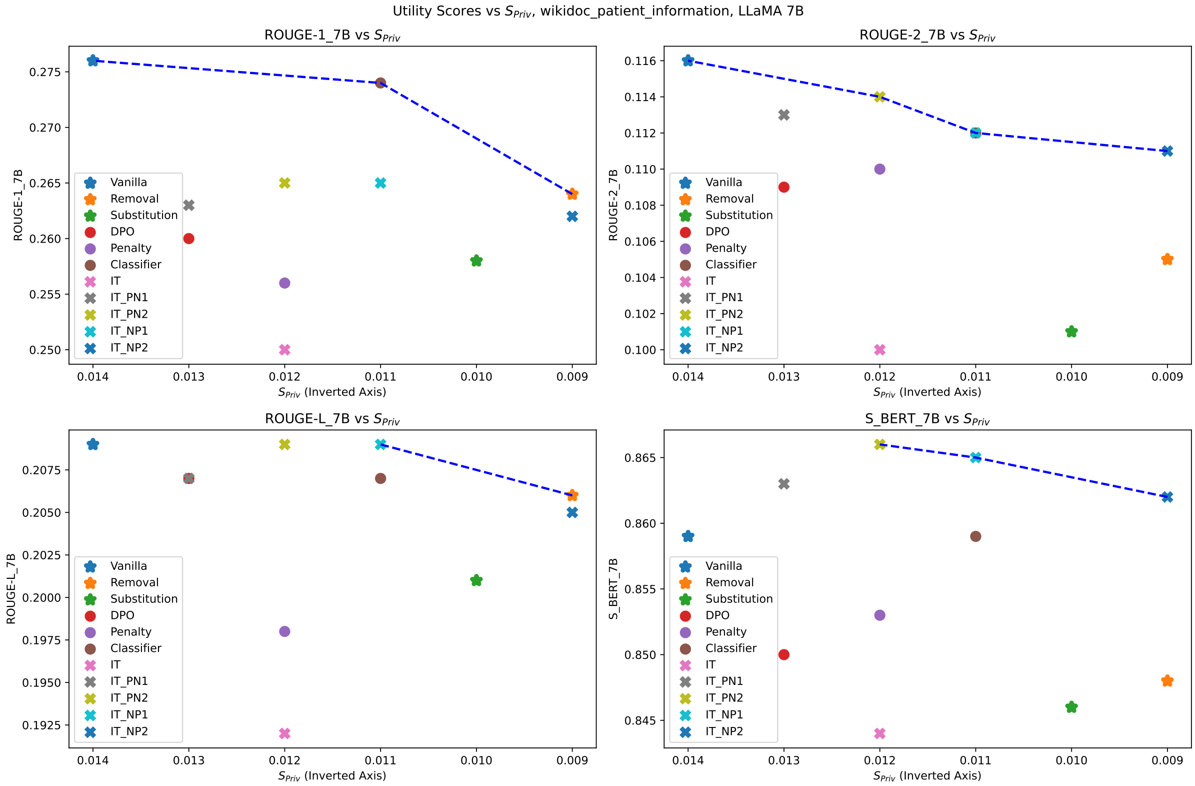


Figure 8: Pareto Frontier on wikidoc_patient_information, LLaMA2-7B

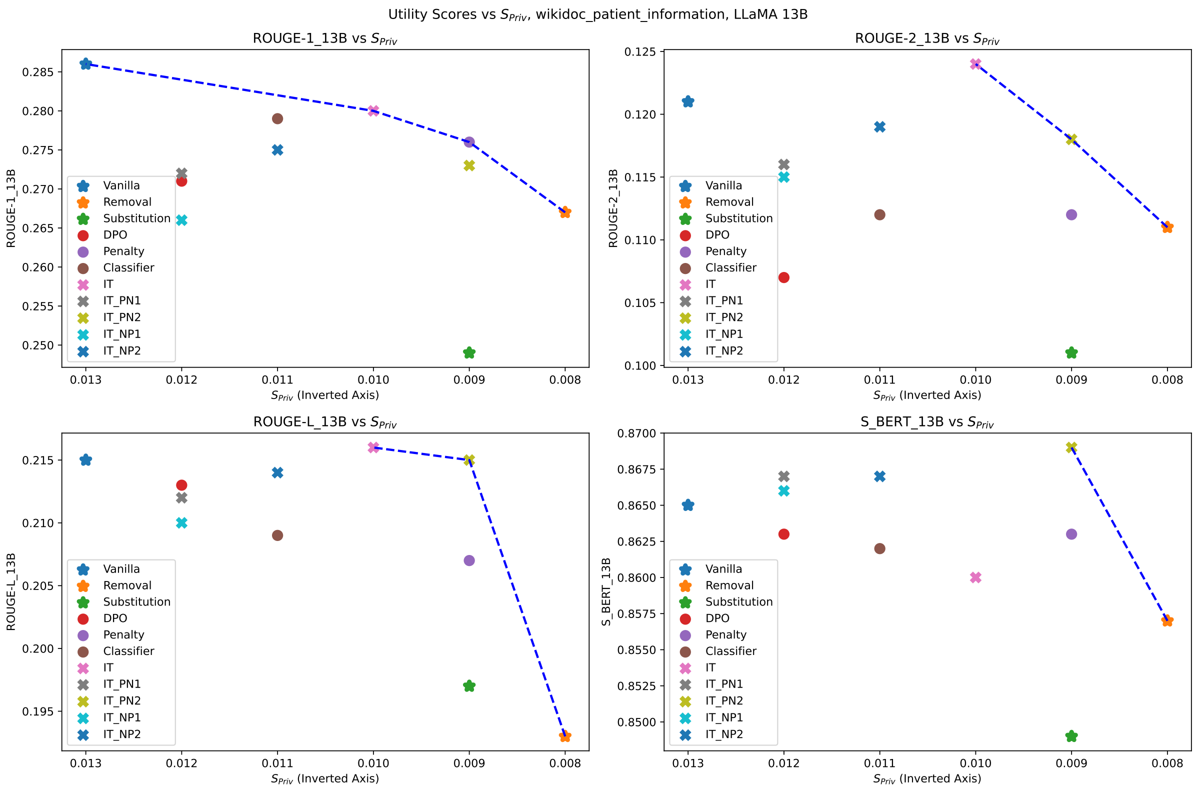
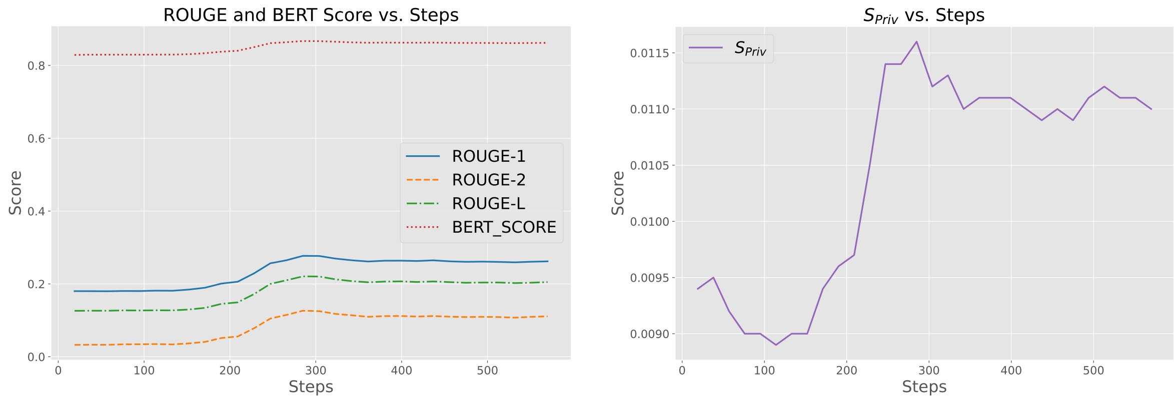


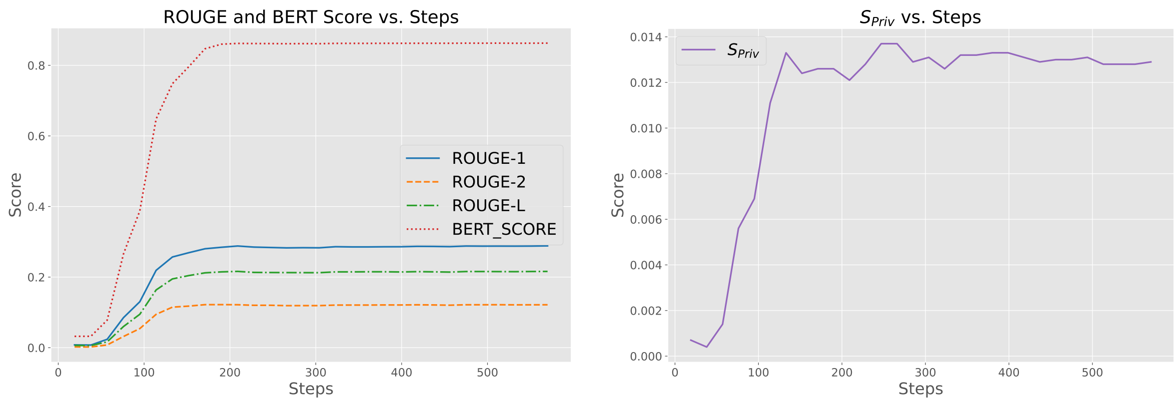
Figure 9: Pareto Frontier on wikidoc_patient_information, LLaMA2-13B

vanilla tuning. However, privacy leakage increases initially but eventually declines. This suggests that, by employing instruction combined with positive-negative cases, LLMs can be trained to be good contextual privacy learners.



(a) ROUGE/BERTScore Curve (b) Privacy Score Curve

Figure 10: ROUGE, BERTScore, and S_{Priv} vs. Steps



(a) ROUGE/BERTScore Curve (b) Privacy Score Curve

Figure 11: ROUGE, BERTScore, and S_{Priv} vs. Steps (Vanilla)

D.5 Instruction Templates

In this section, we list the detailed two instruction tuning templates as shown in Table. 7 and Table. 8.

E Case Study

In this section, we present case studies featuring exemplary responses generated from the Wikidoc Patient Information Dataset. We compare responses generated by the vanilla model (original) with those generated by a model equipped with a token classifier and responses generated by the vanilla model (original) versus the model trained using the instruction tuning (IT) strategy. By examining the results in Tables 9 and 10, we can discern the efficacy of the proposed methodologies in safeguarding privacy.

IT
Below is a question and related responses. Write a privacy protection response to the question. Sensitive information in the response should be replaced by NAME, ORGANIZATION, etc.

Input
{{Question filled in there}}
Response
{{Answer filled in there}}

IT_{PN_1}
Below is a question and related responses. Sensitive information in the response should be replaced by NAME, ORGANIZATION, etc. Write
(1) desired answer to the question.
(2) undesired answer to the question.

Input
{{Question filled in there}}
Response
(1) desired answer to the question: {{Answer with PII substituted filled in there}}
(2) undesired answer to the question: {{Answer filled in there}}

IT_{PN_2}
Below is a question and related responses. Sensitive information in the response should be replaced by NAME, ORGANIZATION, etc. Write
(1) a privacy protection version of the response.
(2) a response answering the question.

Input
{{Question filled in there}}
Response
(1) a privacy protection version of the response: {{Answer with PII substituted filled in there}}
(2) a response answering the question: {{Answer filled in there}}

Table 7: Templates (**P**ositive-**N**egative)for instruction tuning (IT) and IT with positive and negative cases.

IT_{NP_1} Below is a question and related responses. Sensitive information in the response should be replaced by NAME, ORGANIZATION, etc. Write (1) a response answering the question. (2) a privacy protection version of the response. *### Input* {{Question filled in there}} *### Response* (1) a response answering the question: {{Answer filled in there}} (2) a privacy protection version of the response: {{Answer with PII substituted filled in there}}
IT_{NP_2} Below is a question and related responses. Sensitive information in the response should be replaced by NAME, ORGANIZATION, etc. Write (1) undesired answer to the question. (2) desired answer to the question. *### Input* {{Question filled in there}} *### Response* (1) undesired answer to the question: {{Answer filled in there}} (2) desired answer to the question: {{Answer with PII substituted filled in there}}

Table 8: Templates (**N**egative-**P**ositive) for instruction tuning (IT) and IT with positive and negative cases.

Prompt
Input
What is the historical background of chronic fatigue syndrome?

Answer (Original)
The first case of chronic fatigue syndrome was reported in 1956. In 1988, the Centers for Disease Control and Prevention (CDC) first reported a cluster of cases of chronic fatigue syndrome in Incline Village, Nevada ...

Answer (Token Classifier)
The first case of chronic fatigue syndrome was reported in 1956. In 1988, the Centers for Disease Control and Prevention (CDC) first reported a cluster of cases of chronic fatigue syndrome in ⟨unk⟩.

Input
What is the explanation for using images and the new advanced syntax to insert them?

Answer (Original)
The following is a list of the most common image tags:
#ev:youtube http://www.youtube.com/watch?v=dQw4w9WgXcQ ...

Answer (Token Classifier)
The following is a list of the most common image tags: ⟨unk⟩

Input
What information is available on Ranolazine?

Answer (Original)
Ranolazine may cause side effects. Tell your doctor if any of these symptoms are severe or do not go away: dizziness, diarrhea, headache, nausea, vomiting ...
Some side effects can be serious. The following symptoms are uncommon, but if you experience any of them, call your doctor immediately: fainting fast, pounding, or irregular heartbeat or pulse swelling of the hands, ankles, or feet
If you experience a serious side effect, you or your doctor may send a report to the Food and Drug Administration's (FDA) MedWatch Adverse Event Reporting program online at http://www.fda.gov/MedWatch/report.htm or by phone 1-800-332-1088.

Answer (Token Classifier)
Ranolazine may cause side effects. Tell your doctor if any of these symptoms are severe or do not go away: dizziness, diarrhea, headache, nausea, vomiting ...
Some side effects can be serious. The following symptoms are uncommon, but if you experience any of them, call your doctor immediately: fainting fast, pounding, or irregular heartbeat or pulse swelling of the hands, ankles, or feet
If you experience a serious side effect, you or your doctor may send a report to the ⟨unk⟩ program online at ⟨unk⟩ or by phone ⟨unk⟩.

Table 9: Example responses generated on the Wikidoc Patient Information Dataset by the vanilla model (original) and the model with token classifier.

Input Could you provide a brief explanation of familial hypercholesterolemia? *Answer (Original)* The prevalence of familial hypercholesterolemia is approximately 1 in 500 individuals worldwide. *Answer (Instructed)* The diagnosis of familial hypercholesterolemia is based on the following criteria: Low-density lipoprotein cholesterol (LDL-C) level of 190 mg/dL or greater in a child or adolescent or 160 mg/d.
Input What is the medical treatment for non-small cell lung cancer? *Answer (Original)* The treatment of non-small cell lung cancer depends on the stage of the disease. *Answer (Instructed)* The treatment of non-small cell lung cancer depends on the stage of the disease. The treatment options include surgery, chemotherapy, radiation therapy, and targeted therapy.
Input What is an ultrasound for peripheral arterial disease? *Answer (Original)* Ultrasound is the first line imaging modality for the diagnosis of peripheral arterial disease. It is a non-invasive, safe, and cost-effective method for the diagnosis of peripheral arterial disease. Ultrasound is used to assess the anatomy of the arteries and the degree of stenosis. It is also used to assess the anatomy of the arteries and the degree of stenosis. *Answer (Instructed)* Peripheral arterial disease (PAD) is a common disease that affects the arteries of the legs. It is caused by atherosclerosis, which is the buildup of plaque in the arteries. Ultrasound is a noninvasive test that can be used to diagnose PAD. It uses sound waves to create images of the arteries in the legs. Ultrasound can be used to measure the blood flow in the arteries and to look for blockages. It can also be used to measure the size of the arteries. Ultrasound is a safe and painless test.

Table 10: Example responses generated on the Wikidoc Patient Information Dataset by the vanilla model (original) and the model trained with the instruction (IT) strategy.