Protecting Private Information While Preserving Semantic Integrity in LLM-Assisted Systems: It *can* be done.

Anonymous ACL submission

Abstract

With the increasing use of AI-assisted systems, there is growing concern over privacy leaks, especially when users share sensitive personal data in interactions with Large Language Models (LLMs). Conversations shared with these models may contain Personally Identifiable Information (PII) that could be exposed. To address this issue, we present the LOPSIDED[1] framework, a semantically-aware privacy agent designed specifically for remote LLMs. Our approach involves pseudonymizing requests during inference and de-pseudonymizing them once the response is generated, ensuring that sensitive information is protected without compromising the quality of the LLM's output. We evaluate our approach using real-world conversations sourced from ShareGPT. Furthermore, we augment and annotate this data to determine whether named entities are relevant to the prompt and impact the LLM's output. Our analysis reveals that our method reduces utility errors by a factor of 5 compared to baseline techniques, all while maintaining privacy.

1 Introduction

AI-assisted tools are becoming increasingly popular, with users relying on third-party services to complete various tasks, from generating content to analyzing data. These systems operate by processing user inputs, such as text, and leveraging large language models to generate relevant responses. These models typically reside on remote servers, requiring user data to be transmitted for processing. When users input text, they may unknowingly share Personally Identifiable Information (PII), such as names and addresses, raising concerns about privacy and the potential misuse of sensitive data (Aura et al., 2006; Hardinges et al., 2024). For example, in 2023, Samsung employees unintentionally leaked sensitive company information into ChatGPT (Mauran, 2023). Such incidents could lead to unintended data exposure, emphasizing the need for strong privacy safeguards.

Prior work has focused on identifying and mitigating privacy risks in AI-assisted systems by removing PII (Di Cerbo and Trabelsi, 2018; Stubbs et al., 2015). One common approach is pseudonymization, a technique used to protect users' privacy by replacing PII with entities of the same class. For example, a city name like *Chicago* might be substituted with *Los Angeles* to obscure the original data while maintaining the overall structure of the input. However, such techniques can introduce unintended consequences. If a system relies on specific details for accuracy, altering key information may lead to misleading or incorrect results. For instance, if a user asks, "What is the population of Chicago?" and the system modifies it to "What is the population of Los Angeles?", the semantic integrity of the query is compromised. This highlights a key challenge in privacy-preserving techniques — ensuring that user data remains protected without distorting the intended *semantic meaning* of their input.

More recently, large language models (LLMs) have been explored for PII removal in AI-assisted systems (Chen et al., 2023; Dou et al., 2023). For example, Hide and Seek (HAS) (Chen et al., 2023) anonymizes any PII within a prompt before it is transmitted to a cloud-based language model and then de-anonymizes the LLM's response. However, even such techniques may face challenges in preserving the accuracy and context of the output, as sanitizing the prompt without considering semantic meaning could lead to unintended changes in context or produce misleading responses. Thus, a key research question we address in this work is *how to effectively pseudonymize prompts for PII removal while maintaining the semantic integrity of the LLM's response.*

[1]Local Optimizations for Pseudonymization with Semantic Integrity Directed Entity Detection

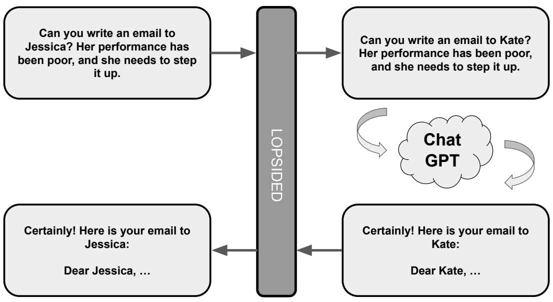


Figure 1: The LOPSIDED privacy agent system design.

Currently, all techniques rely on an *all-or-nothing approach*, where either all private entities are removed, or none are. This method can render the system unusable for users, as they may not be able to interact effectively with the tool if essential information is removed. *Our key insight in this work is to develop a more nuanced approach that selectively sanitizes PII data and replaces it with something that generates a semantically similar response.* For example, if a user asks about the weather in Palo Alto, it could be replaced with San Jose, maintaining privacy while preserving the context and accuracy of the response. This approach ensures that the system produces meaningful outputs, rather than substituting sensitive information with completely unrelated data. Similarly, for other types of PII, we aim to use contextually appropriate replacements that preserve both privacy and the integrity of the user's inquiry.

To address this challenge, we propose LOPSIDED, a lightweight framework that balances PII removal and semantic response preservation. Our work focuses on maximizing user privacy by locally sanitizing sensitive information before it is transmitted to remote LLMs. As shown in Figure 1, the privacy agent operates as an intermediary between the user and the remote cloud. It intercepts user input, sanitizes the prompt by removing or replacing PII, and then processes the response by de-anonymizing it before presenting it to the user. This ensures that privacy-sensitive data is never exposed to external servers while maintaining the relevance of the system's response. We note that there are situations where a replacement could completely alter the meaning. In such cases, we prioritize maintaining the utility of the response while addressing privacy concerns. Moreover, since sanitization must occur locally, we explore the use of smaller models that can be deployed on the user's device to enable efficient privacy protection. Our key contributions are as follows:

LOPSIDED Design: We formulate the problem of semantic-aware privacy for AI-assisted environments, where the goal is to pseudonymize named entity while preserving the utility of the LLM response. To address this problem, we introduce LOPSIDED, a framework which ensures that named entities can be modified without disrupting the semantic integrity of the response, making it both privacy-preserving and semantically accurate.

Semantic-aware Privacy Dataset: We augment the ShareGPT dataset, which contains real-world ChatGPT conversation histories, by annotating named entities to determine whether they are relevant or irrelevant to the prompt. This process results in the creation of a novel 866-sample evaluation dataset, specifically designed for testing semantic-aware privacy agents. This dataset serves as a benchmark for evaluating privacy-preserving techniques, while ensuring that the semantic integrity of AI-generated responses remains intact.

Evaluation and Analysis: We evaluate our technique using real-world conversation prompts from ShareGPT and compare it against several baseline methods, including those fine-tuned on our dataset. Our analysis shows that prior work often prioritizes privacy at the expense of utility. In contrast, our approach reduces utility errors by a factor of 5 while still effectively preserving privacy, demonstrating a significant improvement in balancing both privacy protection and semantic integrity.

2 Background

2.1 Personally Identifiable Information

Personally Identifiable Information (PII) refers to any information that can be used to identify an individual, either directly or indirectly. PII is a key concept in privacy and data protection, as its exposure can lead to identity theft, fraud, and other security risks (Seh et al., 2020; Krishnamurthy and Wills, 2009). The definition of what constitutes PII can vary, but generally, it includes both direct and indirect identifiers (Pilán et al., 2022). Direct identifiers are information that can directly identify an individual on their own, such as names and address. In contrast, indirect identifiers are information that, when combined with other data, can lead to the identification of an individual. Examples of indirect identifiers include a person's job title, gender, and geographic location data. We provide additional details on the PII fields considered in this study in the Appendix section.

2.2 Named Entity Recognition

Prior studies have highlighted that identifying PII is a significant challenge (Nadeau and Sekine, 2009; Pilán et al., 2022). A key challenge is that the definition of PII can change over time (Lukas et al., 2023; Brown et al., 2022). Moreover, as datasets grow larger and more complex, automatically detecting PII becomes increasingly difficult and often requires human annotators for accurate identification. To address these challenges, most techniques rely on Named Entity Recognition (NER), a method used to identify and classify entities such as names, locations, and organizations within text.

Existing methods, such as spaCy (Honnibal and Montani, 2017) and NLTK (Bird et al., 2009), leverage language models to perform NER. For example, spaCy is a popular NLP library that uses pre-trained models to recognize named entities in text. The model identifies entities such as names, locations, organizations, and other relevant categories, classifying them into predefined labels like `[PERSON]`, `[GPE]` (Geopolitical Entity), or `[ORG]` (Organization). Prior work has adopted spaCy as part of their pipeline to identify and anonymize named entities (Chen et al., 2023).

2.3 Related Work

Research has shown that language models can lead to the leakage of PII (Rocher et al., 2019; Vakili and Dalianis, 2021; Huang et al., 2022; Lee et al., 2023). As a result, there has been recent work focused on mitigating these privacy concerns in language models (Li et al., 2021; Shi et al., 2021; Yu et al., 2021; Chen et al., 2023). These mitigation techniques often involve the use of differential privacy guarantees during the training pipeline. Additionally, efforts have been made to reduce PII leakage in language models specifically (Zhao et al., 2022; Lukas et al., 2023; Chen et al., 2023). However, much of this work primarily focuses on privacy, often neglecting the preservation of utility and the semantic integrity of the generated outputs.

Privacy self-disclosure is closely related to PII, but with a focus on the intentional sharing of personal information by individuals (Dou et al., 2023; Valizadeh et al., 2021). Prior work has focused on various types of self-disclosure, including mental health and employment history (De Choudhury et al., 2016; Yates et al., 2017; Tonneau et al., 2022). These studies have explored how individuals manage their privacy when interacting with social media platforms and the risks associated with voluntarily sharing personal details. In contrast, our work focuses on situations where users may inadvertently share personal information with AI-assisted systems, specifically in interactions with large language models. Our approach addresses the potential risks of unintended information leakage while still allowing the AI to generate accurate and contextually relevant responses.

Metric	Test Set	Training Set
# of Prompts	866	2595
# of Entities	1195	3696
Entities per Prompt	1.38	1.42
Avg # of Word Tokens	49.38	49.03
Avg Entity Length	6.80	7.24
Max Ents in a Prompt	8	31
# Prompts Req. Review	30	N/A
Rejections	20	N/A

Table 1: Data statistics and validation summary.

3 Dataset Description

3.1 Data Collection

We use the ShareGPT dataset, the only publicly available dataset, consisting of 70K ChatGPT conversation history of users (Chiang et al., 2023). This dataset includes a wide variety of user-generated conversations with AI systems, some of which contain named entities. We focus only on the first turn of a chat-based interaction in the dataset, as expanding the context to include multiple turns would significantly increase the resources required for training the models. However, our approach is extendable to multi-turn conversations, and future work could explore how to efficiently handle longer context windows while maintaining the same level of privacy and semantic integrity.

The majority of the 70K samples in this corpus do not contain PII or sensitive information. To identify the prompts that do contain PII, we utilize Amazon Comprehend's PII Detection Service[2]. This service is a fully managed machine learning tool that automatically detects personally identifiable information (PII) in text. It identifies sensitive data such as names, locations, and other types of information that can be linked to an individual. After running Comprehend on the dataset, the service flagged 3461 samples as containing PII.

[2]https://docs.aws.amazon.com/comprehend/

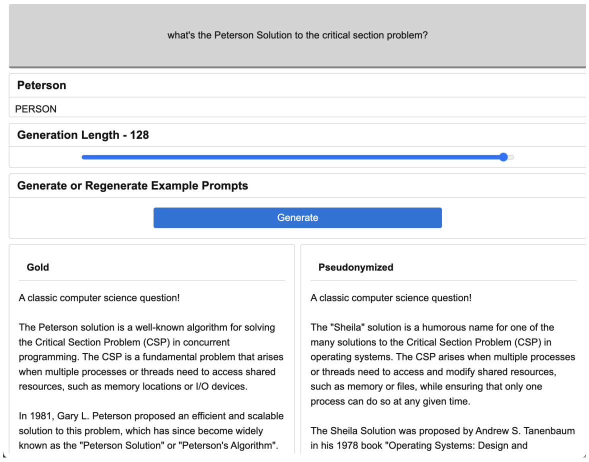


Figure 2: Web interface for data annotation.

Type	Relevant	Irrelevant
Person	228	363
Organization	160	54
Facility	5	1
City/Country	267	23
Landmark	26	4
Demographic	62	2
Total	748	447

Table 2: Statistics of our human annotated dataset.

However, upon manually inspecting these samples, we found that the service is not always accurate and sometimes flags sentences that do not actually contain any PII. Nevertheless, we decided to retain these samples in the dataset. Later, when we use these prompts with GPT to identify named entities, the GPT responses for these flagged prompts contain no named entities, confirming that they do not actually contain PII. Table 1 summarizes the key statistics of our dataset.

3.2 Data Annotation

We begin by tagging the named entities using spaCy, which categorizes each entity (e.g., location, name). For each prompt, we then annotate whether the named entities are relevant or irrelevant. We consider a named entity relevant if substituting it would alter the meaning of the prompt or significantly impact the quality of the response from an LLM. On the other hand, irrelevant named entities can be safely replaced without affecting their meaning or response from an LLM.

To annotate the named entities and assess their relevance to the prompt, we developed a custom web interface designed to streamline the annotation process (see Figure 2). This interface enables annotators to easily tag named entities detected within each prompt and categorize them as either relevant or irrelevant. A local instance of Llama 3 8b runs in the background, allowing users to test how our privacy agent would impact the model's responses. On the left-hand side of the interface, annotators can view Llama's original output without any privacy intervention. On the right-hand side, they can observe the response generated by Llama after replacing the identified entity with a randomly generated pseudonym of the same type. This comparison is provided as guidance to assist annotators in making informed decisions, though it does not serve as the sole criterion for determining relevance. Detailed instructions are available in the Appendix sections.

We recruited three graduate students from our lab, who volunteered to assist with the annotation process. Each volunteer was trained on how to use the interface and was instructed to label the entities in the dataset as relevant or irrelevant for PII. Given the significant resources required for manual annotation, we limited the scope of the data annotation to the test dataset (866 samples). This allowed us to evaluate how our approach performed in preserving privacy and semantic integrity. In total, the annotators spent approximately 6 hours completing the task.

3.3 Data Validation

We validated our data using a majority voting approach. Specifically, if two out of the three annotators agreed on an entity being relevant, then it was classified as relevant; if two out of the three annotators agreed that it was irrelevant, it was considered irrelevant.

Because annotators have the ability to reject prompts, there is a small chance that there is a three-way tie, where the first annotator says that an entity is irrelevant, the second says it is relevant, and the third rejects the prompt. This situation did not arise, but there *were* cases in which one annotator rejected and the others did not. These annotations were subjected to a manual review by the author. As a result, 20 samples were rejected due to inconsistencies or ambiguities in the annotations. Additionally, 30 samples were further revised after a closer examination to ensure their accuracy.

3.4 Data Analysis

Table 2 highlights the key characteristics of the human-annotated dataset. We observe that, for

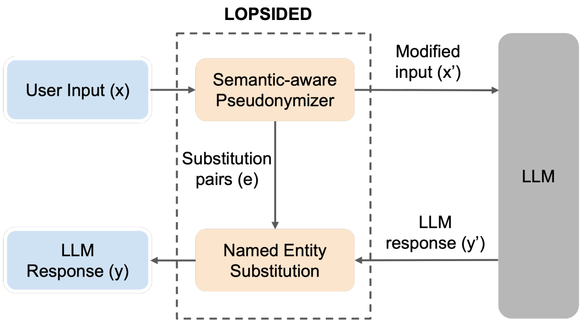


Figure 3: Overall workflow of LOPSIDED framework.

each named entity category, the number of relevant and irrelevant samples varies. In general, relevant tags occur 1.6 times more frequently than irrelevant tags (Table 2), which aligns with the intuition that users typically include information that is relevant to the task. However, 37% of the samples were deemed irrelevant, indicating that these named entities can be safely replaced without affecting LLM's response.

4 LOPSIDED Design

Figure 3 illustrates the overall workflow of the LOPSIDED framework, which consists of two main components: *semantic-aware pseudonymization* and the *named entity substitution* module. Unlike prior methods, the semantic-aware pseudonymization module is designed to generate semantically appropriate replacement entities, referred to as *pseudonyms*, for sensitive information while preserving the meaning of both the input prompt and the response derived from it. Specifically, when the user provides an input prompt, the semantic-aware pseudonymization module identifies and replaces private named entities, ensuring that this does not impact the overall response. The sensitive named entities are then stored locally for later use. The sanitized prompt is subsequently sent to the remote cloud provider, where a response is generated. Once returned, the named entity substitution module utilizes both the locally stored private named entity information and the generated response to produce the final output. As a result, the user receives a response that protects privacy while maintaining the utility of the original prompt.

4.1 Semantic-aware Pseudonymization

This component substitutes named entities within a user prompt with pseudonyms. Formally, let x represent the original input prompt. We train a pseudonymization model $\mathcal{P}$ to generate an output consisting of a modified prompt x' and a set of entity pairs $e = \{(e_{\text{orig}}, e_{\text{pseudo}})\}$, where e_{orig} is a named entity identified in the original prompt x and e_{pseudo} is the corresponding pseudonym or replacement entity used in x'.

Data collection for Pseudonymization. To train the model, we first collect a dataset of sentences containing sensitive named entities. Since manually modifying each sentence is both time-consuming and costly (Wu et al., 2023), we leverage state-of-the-art language models like GPT-4o to automate this process (Liu et al., 2023). Specifically, we prompt GPT-4o to generate semantically appropriate replacement entities for the sensitive information in the sentences, resulting in a modified version of the input prompt. Figure 6 illustrates a sample response from GPT-4o, and we provide the list of instructions to generate the prompt in the Appendix B.2.1. This approach allows us to create a supervised dataset, which we then use to distill the knowledge from GPT-4o into our model. We conducted a manual inspection of the dataset to ensure that the replaced entities were not similar to the original.

Model Training. We train the pseudonymization model $\mathcal{P}$ on the curated dataset $\mathcal{D}_{\text{pseudo}}$ using the following objective:

$$\max_{\mathcal{P}} \mathbb{E}_{(x,e,x') \sim \mathcal{D}_{\text{pseudo}}} \log p_{\mathcal{P}}(e, x'|x) \qquad (1)$$

where $e = \{(e_{\text{orig}}, e_{\text{pseudo}}\})$ is a set of named entity substitution pairs, and x' is the modified prompt containing the pseudonymized entities. Since our primary goal is to run the model locally with lower computational requirements, we opt for the smaller 2B-parameter Gemma 2 model for our experiments (Team, 2024).

4.2 Named Entity Substitution

The key goal of the named entity substitution model S is to reconstruct the original response y as transparently as possible, ensuring it remains semantically similar to the original response that would have been generated from the unmodified input x. By leveraging the stored named entity mappings e, the model S reinserts the original entities into y', the response from the remote LLM. This ensures that the final output maintains both privacy protection and the semantic response of the prompt.

Formally, let x' represent the modified input and y' denote the response generated by the remote

LLM using x'. We train a substitution model S to reconstruct the response y, which would have been generated by remote LLM from the original input x. The model S takes the modified remote LLM response y', and uses a set of named entity mapping $e' = \{(e_{\text{pseudo}}, e_{\text{orig}})\}$ to restore the original entities within a response y.

Data collection for substitution model. We augment the dataset collected during the pseudonymization step by incorporating responses from GPT-4o. For each original input x and its corresponding modified version x', we query GPT-4o to generate both the original response y (for x) and the modified response y' (for x'). Our appendix contains additional information on the structure of the data collected from GPT-4o, including our prompting techniques in Appendix B.2.1. The process results in a dataset consisting of tuples of the form (original input, modified input, original response, modified response, named entity substitution pairs). We then use this dataset to train the substitution model S to reconstruct the original response y from the modified response y' and named entity substitution pairs e'.

Model training. We train the substitution model S on the dataset $\mathcal{D}_{\text{sub}}$ using the following objective:

$$\max_{S} \mathbb{E}_{(y,e',y')\sim\mathcal{D}_{\text{sub}}} \log p_S(y|y', e') \tag{2}$$

where $e' = \{(e_{\text{pseudo}}, e_{\text{orig}})\}$ is a set of pairs that provides the pseudonym and its corresponding named entity.

5 Evaluation

5.1 Baseline Methods

We compare our techniques with the following baseline methods:

Microsoft Presidio (Mendels et al., 2018). This data protection tool focuses on accurately detecting private information in text for anonymization or removal. It prioritizes privacy but does not consider the utility of the entities it removes.

Presidio Anonymizer w/ Replacement. This modification of the Presidio anonymizer assigns numbers to the entities it replaces (i.e., [NAME_1] .) This name is stored as a mapping to the original text, and is replaced by the Presidio Deanonymizer.

Hide-and-Seek (HaS) (Chen et al., 2023). This privacy framework uses a large language model to anonymize and deanonymize prompts to LLM. The model focuses on privacy but does not consider semantic meaning. We use the available pretrained model for our evaluation.

Hide-and-Seek (fine-tuned). We fine-tune the Hide-and-Seek model on our dataset to improve its performance and adapt it to our specific use case.

5.2 Training

We use a pretrained Gemini-2b-it model, consisting of 2 billion parameters, and fine-tuned it on our dataset. We trained the model on 5 epochs using an A6000 GPU. The batch size was set to 4, and the learning rate was 5e-5. For more details, please see Appendix B.

5.2.1 Metrics

For our evaluation, we use BLEU and ROUGE scores to compare the responses from modified and unmodified prompts. In addition, we evaluate the model using the following metrics:

Privacy Errors: are defined as the ratio of irrelevant named entity recognition (NER) samples that were not replaced when they should have been, to the total number of irrelevant NER samples. This metric measures how often the model failed to anonymize or pseudonymize irrelevant entities that should have been replaced to ensure privacy.

Utility Errors: are defined as the ratio of relevant named entity samples that were incorrectly replaced, to the total number of relevant named entity samples. This metric measures how often the model erroneously replaced relevant entities, which could negatively impact the utility and quality of the LLM's response.

6 Results

6.1 Baseline Performance

We begin by comparing our approach to baseline techniques. In our experiment, we modify the prompt and compare the output generated by the privacy agent to the output produced by GPT-4 alone, without any privacy interventions. To evaluate the performance of each approach, we use standard metrics such as ROUGE and BLEU scores, which assess the quality and similarity of the generated responses (Blagec et al., 2022).

Table 3 compares the performance of various privacy agents. As shown, LOPSIDED outperforms other techniques in terms of overall ROUGE and BLEU scores. In general, models that were not fine-tuned exhibit lower performance. Notably, LOPSIDED achieves higher scores, indicating that its

	ROUGE-1	ROUGE-2	ROUGE-L	BLEU-1	BLEU-2	BLEU-3	BLEU-4
LOPSIDED	**0.796**	**0.625**	**0.654**	**0.720**	**0.641**	**0.595**	**0.564**
HaS (finetuned)	0.461	0.226	0.284	0.149	0.108	0.096	0.090
HaS	0.149	0.102	0.125	0.139	0.129	0.124	0.121
Presidio	0.642	0.443	0.487	0.532	0.444	0.397	0.366
Presidio w/ Repl	0.655	0.454	0.497	0.541	0.453	0.405	0.374

Table 3: Baseline performance comparisons. Bolded values are the highest scores.

responses are closer to the ground truth (i.e., the original, unmodified response) compared to other baseline techniques. Additionally, models that have been fine-tuned tend to have lower scores, further highlighting the effectiveness of LOPSIDED in preserving semantic integrity of the LLM response.

Table 4 provides a qualitative comparison of the output generated by different techniques. Specifically, HaS and other baseline methods fail to preserve accurate date and location information, as they indiscriminately substitute all named entities. This often leads to inaccurate responses.

6.2 Privacy and Utility Evaluation

Next, we evaluate the overall performance of our substitution model in balancing privacy and utility. Specifically, we focus on ensuring that relevant named entities (those critical for maintaining the utility of the response) are not replaced, while irrelevant named entities are effectively substituted to protect user privacy. We use *utility* to refer to named entities that are integral to the meaning of the prompt and the remote LLM's response.

For our evaluation, we use the human-annotated test dataset, which contains labels indicating whether each named entity in a prompt is relevant or irrelevant. By comparing the output of the substitution model with these labels, we can measure how well the model maintains the utility of the response by ensuring that relevant entities remain intact, while effectively substituting irrelevant or private entities.

Figure 4 compares the privacy and utility error rates across different techniques. We observe that HaS achieves a low privacy error of 3% because it primarily focuses on substituting all named entities, regardless of their relevance. However, this approach comes at the cost of higher utility errors.

In contrast, LOPSIDED has a slightly higher privacy error of 8%, but it achieves 5$\times$ fewer utility-related errors, demonstrating its ability to selectively preserve relevant entities while still protecting private information. Compared to Presidio,

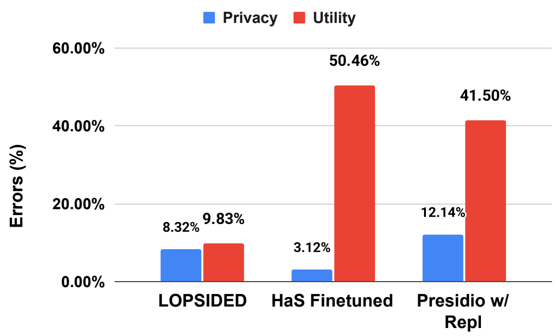


Figure 4: Privacy and utility error comparisons.

LOPSIDED achieves lower errors in both privacy and utility. We also observe that most privacy errors, including those from LOPSIDED, occur when substituting people and organization names. This is because modifying an organization name often alters the context significantly, leading to changes in the LLM's response.

6.3 Text Syntheticity Detection

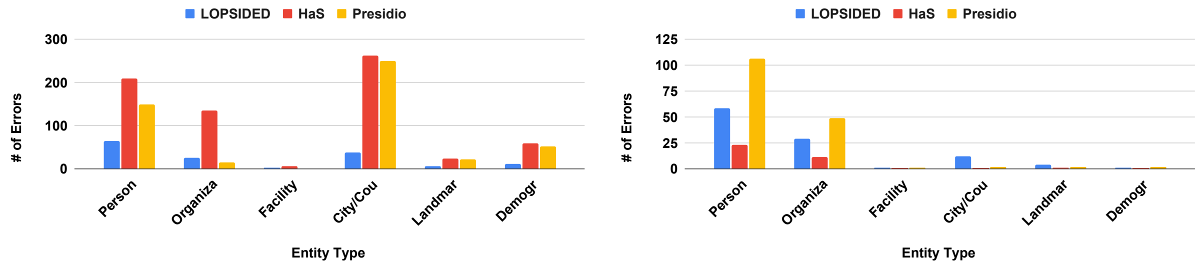
Similar to (Yermilov et al., 2023), we conduct a text syntheticity detection experiment to evaluate whether pseudonymized texts retain similarity to their original versions. This analysis is necessary because pseudonymization can disrupt the relationships between named entities and their surrounding context, potentially leading to inconsistencies in downstream tasks.

To evaluate this, we follow the approach in Yermilov et al., where we combine both pseudonymized and original texts and train a classification model using `bert-base-uncased` (Devlin et al., 2018). to determine whether a given text has been pseudonymized. A high classification accuracy indicates that pseudonymization introduces detectable artifacts, whereas a low classification accuracy suggests that pseudonymized texts closely resemble their original counterparts.

Table 5 presents the text syntheticity classification scores for different techniques. As shown, LOPSIDED achieves the lowest classification

Prompt	When does the sun set in San Antonio mid-summer ?		
Agent	**Privatized Prompt**	**GPT Reply**	**Final Result**
LOPSIDED	... sun set in Houston ...	... typically sets in Houston around 8:30 PM to 8:45 PM CDT ...	... typically sets in San Antonio around 8:30 PM to 8:45 PM CDT ...
HaS Finetuned	... sun set in [GPE] [DATE] ?	To provide ... (GPE) and date (DATE) ... specify the location and the date ...	... sun set in San Antonio mid-summer ... (GPE) and date (DATE) ... specify ...
Presidio w/ Repl	... sun set in <LOCATION_0> <DATE_TIME_0> ?	I can't provide ...	I can't provide ...

Table 4: A sample input ran on each privacy framework, shown at every step of the process.



(a) **Utility related errors**

(b) **Privacy related errors**

Figure 5: Privacy and utility errors by named entity type.

Framework	Detectability (Avg)	Detectability (Final)
LOPSIDED	**46.59%**	**44.88%**
HaS Finetuned	71.84%	83.84%
HaS	85.65%	87.73%
Presidio	60.43%	62.19%
Presidio w/ Repl	51.36%	48.63%

Table 5: Syntheticity detection scores.

score, indicating that its pseudonymized texts are the most similar to the original ones. This suggests that LOPSIDED effectively preserves linguistic and contextual integrity while ensuring privacy.

6.4 Hardware Performance

We also benchmark the performance of our privacy agent running on a laptop to evaluate the real-world feasibility of deploying similar systems. For our evaluation, we used an M3 MacBook Pro with 16GB of memory. We quantized two Gemma 2-based models using `llama.cpp` with a quantization level of q4_k. We observed that the quantized models processed 33 tokens/sec.

Additionally, we evaluated the entire end-to-end process, including queries to a remote LLM. The end-to-end average speed was 15 tokens/sec, though it's important to note that around 33% of this time is attributed to the OpenAI API. Assuming a faster API is used, this speed could approach 20 tokens per second, which is comparable to the original GPT-4's performance[3].

7 Conclusion

LOPSIDED introduces a novel framework for pseudonymizing LLM API prompts while preserving the utility of the user's request. To support this, we present an 866-sample evaluation dataset, validated by human annotators, to assess the effectiveness and utility of privacy agents. This dataset serves as a benchmark for evaluating similar privacy techniques and demonstrates the strengths of our approach. Our results show that LOPSIDED successfully balances both privacy and utility, outperforming other baseline techniques in maintaining semantic integrity while protecting sensitive information. We will release both the dataset and model publicly alongside this paper to foster further research and development.

[3]https://artificialanalysis.ai/models/gpt-4

Limitations

Annotation Resources

Our annotation process was limited to a test set of 866 samples due to the significant effort required for manual annotation. However, LOPSIDED does not depend on the relevant tags in the dataset for training, meaning our training process remains unaffected by the availability of annotated data. That said, we believe that incorporating relevant and irrelevant tags for training could further improve the overall performance of the system. Thus, the development of techniques that leverage these tags could be explored in future work.

Dataset Quality

ShareGPT is a great resource for real world data, but suffers from a lack of quality control. This includes, but is not limited to, nonsense prompts, single word prompts, non-english prompts, and typing/grammar mistakes. To address these issues, we instructed annotators to reject prompts that violated certain guidelines outlined in Appendix A. The rejection rate for the test data was low, and we expect a similar trend in the training set. However, the presence of low-quality samples may have still affected the overall quality of our privacy models.

Ethics Statement

We note that all annotators were graduate students who participated voluntarily, with no compensation provided for their involvement in the project. Their contributions were essential for the successful annotation of the dataset, and we greatly appreciate their efforts in helping to create a valuable resource for future research.

The sensitive nature of private information is heavily considered by the authors. For this reason, we only use data from ShareGPT. Users must opt-in to share their data with this service. No data was collected from users without their knowledge during our work. There are additional risks to be considered with any privacy-related tool. The use of our tool may introduce certain limitations or unintended consequences, as it may occasionally prioritize utility over privacy, particularly when the relevance of certain named entities is critical to the task at hand. This trade-off is inherent in any privacy-preserving approach and highlights the ongoing challenge of balancing privacy protection with maintaining the utility and quality of LLM outputs. We posit that any additional layer in a user's privacy pipeline is a step toward a safer experience when using language model API's.

References

Tuomas Aura, Thomas A. Kuhn, and Michael Roe. 2006. Scanning electronic documents for personally identifiable information. In *Proceedings of the 5th ACM Workshop on Privacy in Electronic Society*, WPES '06, page 41–50, New York, NY, USA. Association for Computing Machinery.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.".

Kathrin Blagec, Georg Dorffner, Milad Moradi, Simon Ott, and Matthias Samwald. 2022. A global analysis of metrics used for measuring performance in natural language processing. *Preprint*, arXiv:2204.11574.

Hannah Brown, Katherine Lee, Fatemehsadat Mireshghallah, Reza Shokri, and Florian Tramèr. 2022. What does it mean for a language model to preserve privacy? In *Proceedings of the 2022 ACM conference on fairness, accountability, and transparency*, pages 2280–2292.

Yu Chen, Tingxin Li, Huiming Liu, and Yang Yu. 2023. Hide and seek (has): A lightweight framework for prompt privacy protection. *Preprint*, arXiv:2309.03057.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Munmun De Choudhury, Emre Kiciman, Mark Dredze, Glen Coppersmith, and Mrinal Kumar. 2016. Discovering shifts to suicidal ideation from mental health content in social media. In *Proceedings of the 2016 CHI conference on human factors in computing systems*, pages 2098–2110.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Francesco Di Cerbo and Slim Trabelsi. 2018. Towards personal data identification and anonymization using machine learning techniques. In *New Trends in Databases and Information Systems*, pages 118–126, Cham. Springer International Publishing.

Yao Dou, Isadora Krsek, Tarek Naous, Anubha Kabra, Sauvik Das, Alan Ritter, and Wei Xu. 2023. Reducing privacy risks in online self-disclosures with language models. *arXiv preprint arXiv:2311.09538*.

Jack Hardinges, Elena Simperl, and Nigel Shadbolt. 2024. We Must Fix the Lack of Transparency Around the Data Used to Train Foundation Models. *Harvard Data Science Review*, (Special Issue 5). Https://hdsr.mitpress.mit.edu/pub/xau9dza3.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. 2022. Are large pre-trained language models leaking your personal information? *arXiv preprint arXiv:2205.12628*.

Balachander Krishnamurthy and Craig E. Wills. 2009. On the leakage of personally identifiable information via online social networks. In *Proceedings of the 2nd ACM Workshop on Online Social Networks*, WOSN '09, page 7–12, New York, NY, USA. Association for Computing Machinery.

Jooyoung Lee, Thai Le, Jinghui Chen, and Dongwon Lee. 2023. Do language models plagiarize? In *Proceedings of the ACM Web Conference 2023*, pages 3637–3647.

Xuechen Li, Florian Tramer, Percy Liang, and Tatsunori Hashimoto. 2021. Large language models can be strong differentially private learners. *arXiv preprint arXiv:2110.05679*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. *arXiv preprint arXiv:2303.16634*.

Nils Lukas, Ahmed Salem, Robert Sim, Shruti Tople, Lukas Wutschitz, and Santiago Zanella-Béguelin. 2023. Analyzing leakage of personally identifiable information in language models. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 346–363. IEEE.

Cecily Mauran. 2023. Whoops, Samsung workers accidentally leaked trade secrets via ChatGPT. (Accessed Feb 2024).

Omri Mendels, Coby Peled, Nava Vaisman Levy, Sharon Hart, Tomer Rosenthal, Limor Lahiani, et al. 2018. Microsoft Presidio: Context aware, pluggable and customizable pii anonymization service for text and images.

David Nadeau and Satoshi Sekine. 2009. A survey of named entity recognition and classification. In *Named Entities: Recognition, classification and use*, pages 3–28. John Benjamins publishing company.

Ildikó Pilán, Pierre Lison, Lilja Øvrelid, Anthi Papadopoulou, David Sánchez, and Montserrat Batet. 2022. The text anonymization benchmark (tab): A dedicated corpus and evaluation framework for text anonymization. *Computational Linguistics*, 48(4):1053–1101.

Luc Rocher, Julien M Hendrickx, and Yves-Alexandre De Montjoye. 2019. Estimating the success of re-identifications in incomplete datasets using generative models. *Nature communications*, 10(1):1–9.

Adil Hussain Seh, Mohammad Zarour, Mamdouh Alenezi, Amal Krishna Sarkar, Alka Agrawal, Rajeev Kumar, and Raees Ahmad Khan. 2020. Healthcare data breaches: Insights and implications. *Healthcare*, 8(2).

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

Amber Stubbs, Christopher Kotfila, and Özlem Uzuner. 2015. Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/uthealth shared task track 1. *Journal of Biomedical Informatics*, 58:S11–S19. Supplement: Proceedings of the 2014 i2b2/UTHealth Shared-Tasks and Workshop on Challenges in Natural Language Processing for Clinical Data.

Gemma Team. 2024. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

Manuel Tonneau, Dhaval Adjodah, João Palotti, Nir Grinberg, and Samuel Fraiberger. 2022. Multilingual detection of personal employment status on twitter. *arXiv preprint arXiv:2203.09178*.

Thomas Vakili and Hercules Dalianis. 2021. Are clinical bert models privacy preserving? the difficulty of extracting patient-condition associations. In *AAAI 2021 Fall Symposium on Human Partnership with Medical AI: Design, Operationalization, and Ethics (AAAI-HUMAN 2021), Virtual Event, November 4-6, 2021*.

Mina Valizadeh, Pardis Ranjbar-Noiey, Cornelia Caragea, and Natalie Parde. 2021. Identifying medical self-disclosure in online communities.

Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A Smith, Mari Ostendorf, and Hannaneh Hajishirzi. 2023. Fine-grained human feedback gives better rewards for language model training. *Advances in Neural Information Processing Systems*, 36:59008–59033.

Andrew Yates, Arman Cohan, and Nazli Goharian. 2017. Depression and self-harm risk assessment in online forums. *arXiv preprint arXiv:1709.01848*.

Oleksandr Yermilov, Vipul Raheja, and Artem Chernodub. 2023. Privacy- and utility-preserving nlp with anonymized data: A case study of pseudonymization. *Preprint*, arXiv:2306.05561.

Da Yu, Saurabh Naik, Arturs Backurs, Sivakanth Gopi, Huseyin A Inan, Gautam Kamath, Janardhan Kulkarni, Yin Tat Lee, Andre Manoel, Lukas Wutschitz, et al. 2021. Differentially private fine-tuning of language models. *arXiv preprint arXiv:2110.06500*.

Xuandong Zhao, Lei Li, and Yu-Xiang Wang. 2022. Provably confidential language modelling. *arXiv preprint arXiv:2205.01863*.

A Annotation

Annotators were requested to reject any prompts that fall into the following categories: (i) Non-english language prompt, (ii) Sexual, violent, or harmful content, and (iii) Single or few word prompt. Most categories were straightforward to annotate, with each prompt typically containing one or two named entities. Based on our observations, the availability of responses from a llama LLM for various substitutions also helped us understand whether a named entity was relevant, which in turn simplified the data annotation process. However, while we did not conduct a qualitative survey on the effectiveness of LLM response in annotation, it is important to note that the annotation process was still influenced by certain subjective judgments made by the annotators. This led to some inter-annotator disagreements. However, we observed that these disagreements were infrequent, with a total of 30 disagreements across the dataset.

In addition to a demo of our web interface to the annotators, we also provide a annotation guide. Below is a direct sample excerpt from the annotation guidelines provided to our annotators.

Annotation Guidelines

A tag is relevant if the word's **meaning** is absolutely required in order to give an acceptable response. Names are usually not required and could be removed or changed to protect a user's privacy - thus, they are not relevant.

The **context** of a tag determines how relevant it is. If the prompt was *"Write an acrostic poem for John"*, that name would be extremely relevant since the output would be completely wrong if we changed or removed the name.

For reference, two example model outputs will be generated to show how changing the token might impact the response. These are meant for reference and should not be the sole decision factor.

Examples of relevant tags:
Show me a list of restaurants in Philadelphia.
Write a song about Caroline.
Note: *Caroline is relevant as songs, like poems, involve rhyming.*

Examples of irrelevant tags:
Write an email firing Laura for not showing up today.
Is it too late to get a passport for my trip to Germany?
Note: *Germany is irrelevant here since passport processing times are not based on destinations.*

Parameter	Value
epochs	5
batch size	4
weight decay	0.001
learning rate	5.00E-04
sequence length	1024
quantization	4bit

Table 6: Hyper parameters for training the replacement and privacy models. HaS Finetuned was also trained using these parameters, but with a batch size of 2.

Parameter	Value
epochs	5
batch size	8
weight decay	0.01
learning rate	2e-05

Table 7: Training parameters for our BERT based syntheticity detection evaluation model.

B Training Details

Table 6 shows the training setup for our models. Since both the privacy and replacement models share the base of Gemma 2 2b-it, there were no modifications required.

B.1 Syntheticity Detection Model Training

B.1.1 Model Hyper-parameters

We provide the training parameter details for our syntheticity detection in Table 7.

B.1.2 Model Data

The syntheticity model was trained on a 60-40 train/evaluation split of the LOPSIDED test data. The classifier was trained to predict a label of either synthetic or not. As input, it is given the original prompt and the response from either the privacy model or from GPT-4o directly.

B.2 Prompt Templates

B.2.1 Pseudonymizer Prompt

The pseudonymizer system prompt is shown in Listing 2, and details what aspects of user privacy we ask the teacher model to consider.

B.2.2 Substitution Prompt for GPT

The system prompt used for the replacement model, and its teacher model, are shown in Listing **??**. This task is considerably simpler, as we do not have the model consider user privacy or semantic meaning.

```
You are PrivacyGPT. You will anonymize the user's prompt while maintaining the
    meaning whenever possible.

Your task is to revise the user's prompt. Your goal is to reword and change all
    private entities that are not strictly relevant to the text. You can change any
    names, places, organizations, etc as long as they wont effect the response when
    changed back.

Remember the following criteria:
* The meaning of the prompt **must not change**.
* Any replacement of names, places, organizations, etc must be semantically similar
    to their original values.
    * Don't use placeholders like [NAME]. Opt for similar entities, such as names
        with the same gender, organizations in the same field, etc.
    * We will replace these replacements again when their response is completed, so
        the user does not notice the effect.
* If the private entities are crucial to the meaning of the prompt then they must
    stay as they appear.
    * For example, a location may remain in the prompt if it is absolutely needed to
         create a response and a replacement would not work.
    * Works of literature often do not rely on the entity remaining the same, but
        there are exceptions, for example if the user requests a rhyming poem or
        song.
* You are **maximizing the privacy** of the user, and **minimizing the effect on
    their request's reponse**.
* If there are no changes, the array of changed entities may be empty, but still
    include the prompt as the "modified_prompt"

You will return your reasoning for each change as well as the change itself. At the
    end, provide the fully modified prompt.

**REMEMBER: ONLY REPLACE THE WORD/TOKEN IF IT WONT CHANGE THE ANSWER OR RESPONSE OF
    THE QUESTION OR TASK.**
Here is the prompt:

{prompt}
```

Listing 1: Pseudonymizer Prompt Template.

```
You are ReplaceGPT, an entity replacement model. Your task is to take an input, and
    output a transformed response that replaces all of the entities specified.

The goal is to minimize the impact of changing the entities. The user should not be
    able to tell this transformation happened.

The user will provide JSON input of the original text, and a list of the entities
    that must be changed.

You will provide a json output that contains the modified text, and a rationale as
    to why you made the changes you made.

Do not make any unnecesary changes that effect the semantic quality of the text, the
     meaning should stay the same.

Only the entities themselves should change, not the meaning.
```

Listing 2: Substitution Prompt Template.

B.3 Additional Data Collection Details

Data from the teacher model are returned via structured JSON format, which is mandated by a schema we provide. Examples of the output are shown in Figures 6 and 7 for the pseudonymizer and replacement pipelines respectively.

```
{
    "changed_entities": [
        {
            "explanation": "The name 'Raven' can be any dog name and doesn't affect
                the story's meaning.",
            "original_entity": "Raven",
            "new_entity": "Shadow"
        },
        ...
    ],
    "modified_prompt": "Write a short story about Shadow...."
}
```

Figure 6: GPT-4o pseudonymization output.

```
{
    "rationale":"The names 'Nyla' and 'Ian' were replaced with 'Raven' and 'Jayson
        '...",
    "modified_output":"Once upon a time .."
}
```

Figure 7: GPT-4o response output for a modified input.