PII-Bench: Evaluating Query-Aware Privacy Protection Systems

Anonymous ACL submission

Abstract

The widespread adoption of Large Language Models (LLMs) has raised significant privacy concerns regarding the exposure of personally identifiable information (PII) in user prompts. To address this challenge, we propose a query-unrelated PII masking strategy and introduce PII-Bench, the first comprehensive evaluation framework for assessing privacy protection systems. PII-Bench comprises 2,842 test samples across 55 fine-grained PII categories, featuring diverse scenarios from single-subject descriptions to complex multi-party interactions. Each sample is carefully crafted with a user query, context description, and standard answer indicating query-relevant PII. Our empirical evaluation reveals that while current models perform adequately in basic PII detection, they show significant limitations in determining PII query relevance. Even state-of-the-art LLMs struggle with this task, particularly in handling complex multi-subject scenarios, indicating substantial room for improvement in achieving intelligent PII masking.

1 Introduction

Recent years have witnessed the widespread adoption of Large Language Models (LLMs), with an increasing number of users directly interacting with these models through APIs for various tasks, ranging from daily conversations to complex analytical work (Sun et al., 2023; Yang et al., 2024b; Wong et al., 2023). Despite the convenience these services offer, users often overlook a significant privacy risk: the prompts submitted to LLMs frequently contain substantial personally identifiable information (PII) (Achiam et al., 2023). Such information is vulnerable not only to interception by malicious actors during transmission (Parast et al., 2022) but also to potential misuse by unethical service providers who might collect and incorporate it into subsequent model training, leading to permanent privacy breaches (Liu et al., 2023).

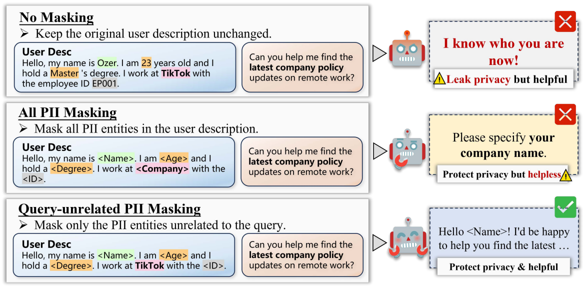

Figure 1: The overall performance of three PII Masking strategies: No Masking, All PII Masking, and Query-unrelated PII Masking. Effective Privacy Protection Systems are required to maintain LLMs' functionality while protect user's privacy as much as possible.

Current practices reveal that the vast majority of users adopt a zero-protection approach when utilizing LLM services, submitting original prompts containing PII directly to the LLMs. While an obvious protection strategy would be to mask all PII (Nakamura et al., 2020; Biesner et al., 2022;Lukas et al., 2023), as shown in Fig. 1, this approach significantly compromises service quality. An ideal Privacy Protection System should maintain LLMs' functionality while maximizing user privacy protection. For instance, when a user inquires about a candidate's suitability for a senior researcher position, masking their educational background and work experience would render the LLM incapable of making an effective assessment.

This observation motivates our proposal of a query-unrelated PII masking strategy: Masking only the PII irrelevant to user queries while retaining essential information. In the aforementioned example, this approach would preserve the candidate's educational and professional information while masking unrelated personal details such as contact information.

The implementation of query-unrelated PII masking stragety faces two-tier challenges. The first involves accurate identification of all PII within the prompt, serving as foundational work.

The second requires determining the relevance of identified PII to user queries. While existing research has made progress in basic PII detection, systematic studies considering query relevance remain scarce.

To advance the field of privacy-preserving language models, we present PII-Bench, a comprehensive evaluation framework designed to assess Privacy Protection Systems' efficacy in preserving Large Language Models' core functionalities while optimizing user privacy safeguards. PII-Bench comprising 2,842 carefully designed test samples across 55 fine-grained PII categories, ranging from basic personal information to complex social relationship data. Each sample consists of three key components: (1) A user query simulating real-world information needs. (2) A context description containing diverse PII. (3) A standard answer indicating query-relevant PII and masking requirements.

Our experimental analysis reveals that while existing models, including Bidirectional Long Short-Term Memory with Conditional Random Fields (BiLSTM-CRF) (Chen et al., 2017), perform adequately in basic PII detection, they demonstrate notable limitations in determining PII query relevance. Even state-of-the-art LLMs face challenges in this task, indicating substantial room for improvement in achieving intelligent PII masking. Despite the recent advances in model architecture and training techniques, Small Language Models (SLM) still show considerable performance gaps compared to larger LLMs, particularly in determining PII query relevance.

The primary contributions of this work include: 1. The first proposal of query-unrelated PII masking strategy, offering novel approaches to maintain LLM service quality while protecting privacy. 2. Development of PII-Bench evaluation framework, enabling systematic assessment of models' capabilities in PII identification and query relevance determination. 3. Experimental revelation of current model limitations in this task, providing direction for future research.

2 Related Work

2.1 Privacy-Preserving Text Processing

Text privacy protection has emerged as a critical challenge in natural language processing applications. Papadopoulou et al. (2022) proposed text sanitization that combines entity detection with privacy risk assessment to guide masking decisions. Shen et al. (2024) extended this approach with an end-to-end framework that preserves task utility during privacy protection. Exploring information preservation, Meisenbacher and Matthes (2024) introduced differential privacy techniques for text modification, demonstrating improved semantic retention over traditional masking methods. While these approaches have advanced privacy protection techniques, they primarily focus on document-level sanitization without considering the dynamic nature of user interactions. Our work introduces query-aware privacy protection that adaptively balances information utility with privacy requirements.

2.2 Query-Aware PII Detection

Traditional PII detection has evolved from rule-based systems (Ruch et al., 2000; Douglass et al., 2005) to neural architectures (Deleger et al., 2013; Dernoncourt et al., 2017; Johnson et al., 2020), with recent work demonstrating the effectiveness of transformer-based models in identifying sensitive information (Asimopoulos et al., 2024). Large language models have shown promising results in recognizing diverse PII types (Singhal et al., 2024; Bubeck et al., 2023), yet they treat all sensitive information with uniform importance. Our framework introduces a novel dimension to PII detection by incorporating query relevance assessment. Rather than applying uniform protection measures, we focus on identifying which PII elements are essential for addressing user queries. This approach enables more nuanced privacy protection by distinguishing between query-related and query-unrelated sensitive information, though the actual masking or protection mechanisms are left to downstream applications.

2.3 Privacy Protection Benchmarks

Existing benchmarks for evaluating privacy protection methods have primarily focused on general PII detection capabilities. Pilán et al. (2022) introduced TAB, a benchmark based on legal court cases, which evaluates text anonymization performance. However, it does not assess the model's ability to distinguish query-related information. The recent work by Sun et al. (2024) proposed evaluation metrics for privacy-preserving prompts, but their focus remains limited to general desensitization effectiveness. Li et al. (2024) developed LLM-PBE to assess privacy risks in language models, though their emphasis is on model-side privacy

Symbol	Description
p	A prompt consisting of a user description and a query
d	User description containing personal information
q	User query specifying the information need
d'	Modified description with masked PII
p'	Modified prompt (d', q) after PII masking
$\mathcal{S}$	Set of subject individuals mentioned in the description
s_i	The i-th subject individual
$\mathcal{E}$	Complete set of PII entities in the prompt
$\mathcal{E}_i$	Set of PII entities associated with subject s_i
e^i_j	The j-th PII entity of subject s_i
$\mathcal{E}_q$	Subset of PII entities necessary for answering query q
$\mathcal{T}$	Set of predefined PII types

Table 1: Notation used throughout in Task Definition.

rather than input text protection.

PII-Bench addresses these limitations by providing a comprehensive evaluation framework that assesses both PII detection accuracy and the ability to determine query-related information. This dual focus enables more realistic evaluation of privacy protection systems in interactive scenarios, where the relevance of sensitive information varies with user queries.

3 PII-Benchmark

3.1 Task Definition

Privacy Protection Systems target at maintaining LLM functionality while maximizing user privacy protection. Let p be a prompt consisting of a user description d and a query q. The description d contains information about multiple subject individuals $\mathcal{S} = \{s_1, ..., s_m\}$. For each subject s_i, there exists an associated set of PII entities $\mathcal{E}_i = \{e^i_1, ..., e^i_k\}$. The complete set of PII entities in prompt p is defined as $\mathcal{E} = \bigcup_{i=1}^{m} \mathcal{E}_i$, where each entity $e \in \mathcal{E}$ belongs to a predefined PII type from set $\mathcal{T}$ (see Appendix A.2). Let $\mathcal{E}_q \subseteq \mathcal{E}$ denote the subset of PII entities that are necessary for answering query q.

Based on this definition, we propose three fundamental evaluation tasks for Privacy Protection Systems:

(1) PII Detection Task: Given prompt p, the model needs to: identify the minimal text spans for all PII entities $e \in \mathcal{E}$; establish associations between each entity e and its corresponding subject $s \in \mathcal{S}$; assign the correct PII type $t \in \mathcal{T}$ to each entity e.

(2) Query-Related PII Detection Task: Given prompt p, the model needs to determine the minimal subset of PII entities $\mathcal{E}_q \subseteq \mathcal{E}$. This subset should only contain PII entities necessary to answer query q, maximizing protection of non-relevant personal information.

(3) Query-Unrelated PII Masking Task: This task is what we propose the optimal form of privacy protection system. Given prompt p, the model should generate a modified description d' where query-unrelated PII entities are masked while preserving the necessary ones. Formally, the model should identify $\mathcal{E}_q$ and generate d' where all PII entities in $\mathcal{E} \setminus \mathcal{E}_q$ are masked while preserving those in $\mathcal{E}_q$. The masking operation should maintain text coherence and readability while ensuring effective privacy protection for non-essential personal information. The resulting prompt $p' = (d', q)$ should enable LLMs to accurately address the query while minimizing exposure of irrelevant personal information.

3.2 PII-Bench Construction

Based on the task definition above, we designed an automated process for constructing the PII evaluation dataset, as illustrated in Fig. 2.

3.2.1 PII Entity Generation

Following Papadopoulou et al. (2022), we expanded the PII type set $\mathcal{T}$ into 55 subcategories (see Appendix A), employing two complementary strategies for entity generation:

(1) Rule-based Generation: Applicable for deterministic PII types with fixed formats or enumerable value sets, such as phone numbers, email addresses, and standardized ID numbers. (2) LLM-based Generation: Applicable for non-deterministic PII types requiring contextual understanding and real-world knowledge, such as occupation descriptions and detailed addresses. This method leverages GPT-4-0806 to generate semantically appropriate and contextually relevant entities.

3.2.2 User Description Generation

Single-Subject Description Construction: The construction of single-subject descriptions follows a three-stage process:

(1) Entity Selection: For subject s, randomly sample n entities ($4 \leq n \leq 16$) from different PII types to construct entity set $\mathcal{E}$. The sampling process ensures diversity of PII types while considering their natural distribution in real-world scenarios. (2) Consistency Optimization: Ensure logical compatibility among entities in $\mathcal{E}$ through designed verification rules. For example, verifies reasonable correspondence between age and educational history as shown in Fig. 2. (3) User Desc Generation:

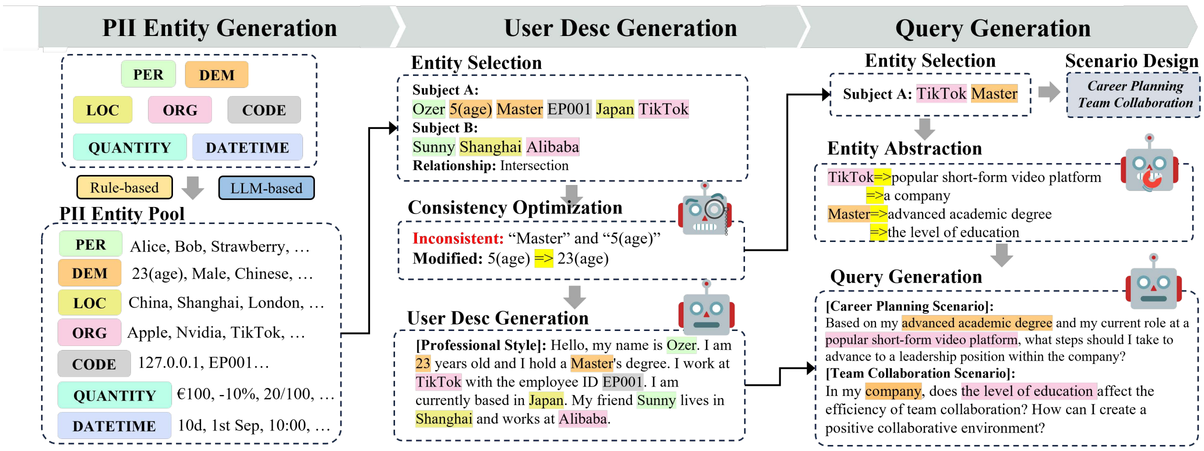

Figure 2: **PII-Bench** synthesis process consists of three main modules: (a) PII Entity Generation, (b) User Description Generation, and (c) Query Generation.

Selects appropriate expression styles to generate the user description. It employs formal description formats like job resumes and employee records in professional scenarios; casual expressions like personal profiles and self-introductions in social scenarios.

Multi-Subject Description Construction: The construction process for multi-subject related descriptions includes these key steps:

(1) Entity Selection: Construct relationship network $R(s_i, s_j)$ for subject pairs (s_i, s_j). Relationship types include intersection relationships like colleagues and alumni, hierarchical relationships like parent-child and teacher-student, and non-intersection relationships with no direct connection. (2) Consistency Optimization: This stage first establishes entity dependency rules based on relationship type R. Then ensures consistency of shared attributes among related subjects, such as company address for employees of the same company. This stage also derives related attributes based on relationship type, such as age differences in parent-child relationships. And finally remove the sample which contains contradictions. (3) User Desc Generation: This stage designs natural interaction environments matching relationship characteristics, placing subjects in realistic scenarios (like meetings, family activities) and constructing multi-party dialogue flows to reflect interactive relationships.

3.2.3 Query Construction

For each description d, query construction follows a four-phase process:

(1) Entity Selection: Randomly sample k entities ($1 \leq k \leq 3$) from $\mathcal{E}$ to form query-relevant entity set $\mathcal{E}_q$. (2) Scenario Design: Construct query contexts that align with real-world application scenarios. The goal is to simulate actual user needs for PII information in specific situations. For example, when $\mathcal{E}_q$ contains "Work Experience": "5 years as Machine Learning Engineer", "Education Background": "Stanford University Ph.D. in Computer Science", this stage generates query scenarios like "As a hiring manager, I need to verify if this candidate's education and relevant work experience meet the requirements for the Senior Researcher position". (3) Entity Abstraction: Map specific PII entities in $\mathcal{E}_q$ to abstract representations, maintaining basic semantic properties while enhancing privacy protection. (4) Query Generation: Integrate abstract entities into corresponding scenarios through GPT-4-0806 model to generate natural queries q that fit practical application scenarios.

3.2.4 Human Verification

All content generated by GPT-4-0806 undergoes rigorous verification by five professional annotators and the authors, focusing on: (1) Completeness and accuracy of PII entity annotations in description d. (2) Correspondence between query q and query-relevant entity set $\mathcal{E}_q$. (3) Overall semantic coherence and scenario authenticity. Complete annotation guidelines and quality control procedures are detailed in Appendix E.

3.3 Dataset Partitioning and Statistics

Table 3 presents the partition and key statistics of PII-Bench, which comprises two main datasets (PII-single and PII-multi) and two specialized test sets (PII-hard and PII-distract). Each sample follows a consistent JSON structure containing four key components: user description, query, comprehensive PII entity annotations, and query-relevant PII labels, as illustrated in Fig. 3.

PII-Single and PII-Multi: Based on the number of subjects in descriptions, the dataset is divided into two main subsets. PII-Single contains 2000 description-query pairs involving single subjects, focusing on model performance in handling individual information. PII-Multi contains 2000 description-query pairs involving multiple related subjects, evaluating model capability in handling privacy information within complex interpersonal networks.

Test-Hard Construction: Select 200 challenging instances from PII-Single and PII-Multi to construct Test-Hard dataset, based on criteria including: (1) Maximum character length of description text d. (2) Highest PII entity density ($|\mathcal{E}|/|d|$). (3) Samples with the most query-relevant entities ($|\mathcal{E}_q|$).

Test-Distract Construction: Construct 200 samples simulating complex multi-user interaction scenarios. Each sample integrates five different descriptions $\{d_1, ..., d_5\}$ from PII-Single and PII-Multi, and constructs queries q involving three of these descriptions based on professional networks, knowledge platforms, and community forum interaction templates. The generation process employs specific dialogue flow transformation strategies to ensure natural transitions and semantic coherence between multiple descriptions. Scenario design particularly emphasizes simulating real-world information interference and complex interaction patterns.

3.4 Human Performance

To establish a human baseline for PII-Bench, we recruited 25 graduate students specializing in data security from top universities across China. All participants had at least two years of research experience in privacy protection and information security. Before the formal evaluation, participants completed a comprehensive training session and passed a qualification test (detailed in Appendix C).

We designed two evaluation sets: a main test set comprising 400 randomly sampled instances (200 each from PII-single and PII-multi), and a challenging set of 100 instances from PII-distract. Each instance underwent independent assessment by five participants through our online evaluation platform. Participants performed two sequential tasks: PII recognition, which involved determining minimal text spans, associated subjects, and PII types for all entities in the user description, followed by query-relevant PII detection to identify entities essential for addressing the given query. The result of the human baseline is shown in Table 2.

Dataset	PII-F1	Query-F1
PII-single	97.2 ± 1.1	95.1 ± 1.3
PII-multi	95.4 ± 1.2	94.3 ± 1.5
PII-hard	91.3 ± 1.1	90.3 ± 1.2
PII-distract	92.8 ± 1.8	91.5 ± 2.1

Table 2: Human performance in PII-Bench. Desc-F1 measures accuracy in the PII recognition task while Query-F1 evaluates the query-relevant PII detection task.

Name	#Sample	Avg #Subject	Avg #Char (Desc)	Avg #PII (Desc)	Avg #Char (Query)	Avg #PII (Query)
PII-single	1,214	1.0	893.48	7.67	211.21	1.95
PII-multi	1,228	2.0	652.65	13.14	236.21	2.06
PII-hard	200	1.5	778.03	10.60	222.09	2.10
PII-distract	200	7.5	4,403.64	51.08	859.69	5.82
All	**2,842**	**1.92**	**1,028.32**	**13.30**	**268.41**	**2.28**

Table 3: Statistic information of PII-Bench.

4 Experiments

4.1 Overall Setup

Traditional Model Baselines: We implemented **BiLSTM-CRF** as a traditional sequence labeling baseline, following the architecture proposed by Huang et al. (2015). We trained the model using Adam optimizer with a learning rate of 1e-3 and batch size of 32 for 50 epochs on the PII-Bench training set.

LLM Baselines: The evaluation encompassed both API-based and open-source large language models. API-based models included GPT-4o-2024-0806 (**GPT4o**) (OpenAI, 2024), Claude-3.5-Sonnet (**Claude3.5**) (Anthropic, 2024), and DeepSeek-Chat **DeepseekV3** (Liu et al., 2024), accessed through their respective official APIs between January 1 and February 10, 2025. Open-source alternatives comprised Llama-3.1-70B-Instruct (**Llama3.1**) (Dubey et al., 2024), and Qwen-2.5-72B-Instruct (**Qwen2.5**) (Yang et al., 2024a).

SLM Baselines: To investigate scaling effects, we included two small-scale language models: Llama-3.1-8B-Instruct (**Llama3.1-SLM**) and Qwen-2.5-7B-Instruct (**Qwen2.5-SLM**). All experiments utilized default parameters with temperature set to 0 to ensure reproducibility. Additionally, we conducted a comprehensive evaluation on smaller, deployment-ready models (0.5B-3B parameters) to assess their viability for on-device PII protection, with detailed results presented in Appendix D.3.

Prompt Baselines: The assessment incorporated multiple prompting strategies for query-related PII

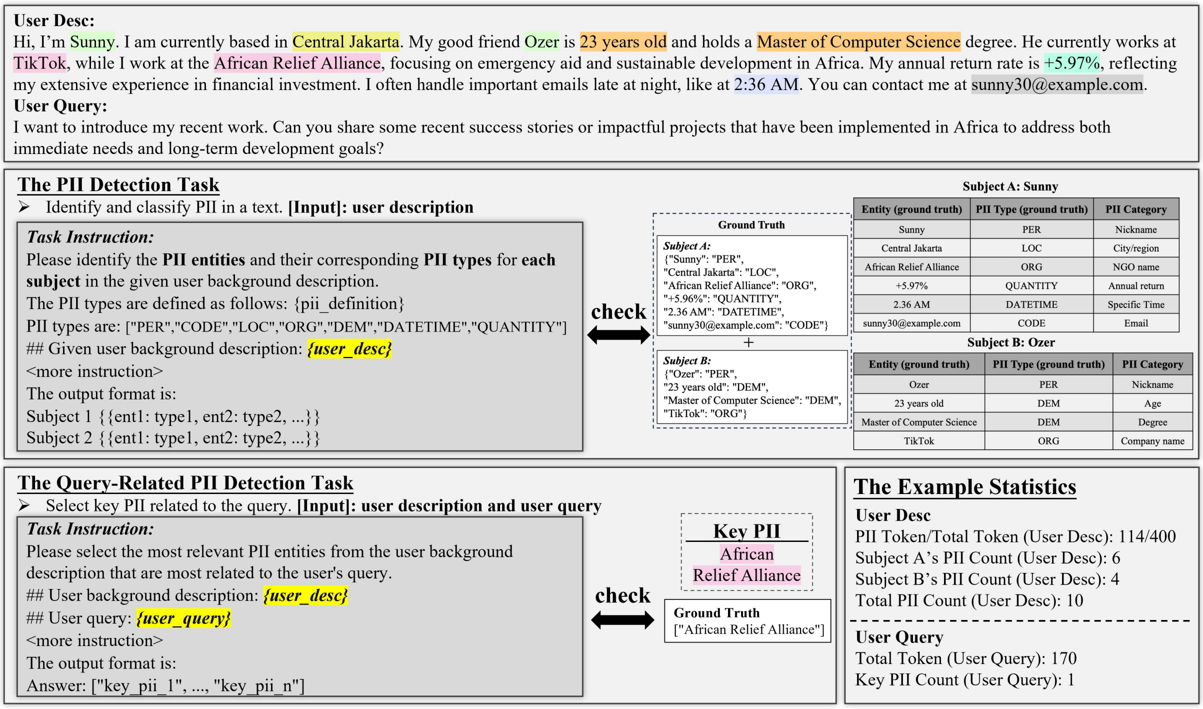

Figure 3: An example from **PII-Bench**, which aims to evaluate Privacy Protection System's ability by masking maximize PII while maintaining LLM's functionality. The evaluation is seperated by two fundamental tasks: (a) The PII Detection Task: Identify and classify PII entities for each subject in the prompt, with ground truth labels shown on the right side. (b) The Query-Related PII Detection Task: Determine which PII entities are necessary for answering the user query, enabling selective masking of irrelevant personal information.

detection. **Naive** inputs the user description and query. **Naive /w Choice** includes a list of candidate PII entities to constrain the selection space. **Self-CoT** (Wei et al., 2022) incorporating step-by-step reasoning prompts. **Auto-CoT** (Zhang et al., 2022), which automates the generation of chain-of-thought demonstrations through three-shot setting. **Self-Consistency (SC)** (Wang et al., 2022), which synthesizes multiple reasoning paths to derive the final output. **Plan-and-Solve CoT (PS-CoT)** (Wang et al., 2023) develops a strategic plan before executing the solution process. Appendix D.5 provides details of each prompts.

Metrics: The PII detection task evaluates model performance through two sets of metrics: **Strict-F1** measures the accuracy of subject identification, entity span detection, and PII type classification simultaneously. **Ent-F1** focuses on entity span detection independent of subject attribution and type classification. For query-related detection, model performance is measured through **Precision**, **Recall**, and **F1**. Considering the inherent variation in entity expressions and potential partial matches, **RougeL-F** is employed for both tasks to complement the exact matching metrics. Detailed computation procedures are provided in Appendix D.1.

4.2 Performance on Query-Unrelated PII Masking

We evaluate models' performance on the query-unrelated PII masking task, which requires both accurate PII detection and relevance assessment. Table 4 presents our results:

Joint task yields improved performance. Models achieve higher F1 scores in this combined task compared to individual query-relevance tasks. GPT4o reaches 0.77 F1 with Self-Consistency prompting, suggesting complementary signals from the joint objective. Open-source models demonstrate comparable capabilities, with both Llama3.1 and Qwen2.5 achieving 0.76 F1 using Auto-CoT.

4.3 Performance on PII Detection

Results on the PII detection task (Table 6) reveal:

Large Language Models demonstrate superior detection capabilities. API-based LLMs achieve strong performance, with DeepSeekV3 and GPT4o leading in Strict-F1 scores (0.903 and 0.891 on PII-Single and PII-Multi, respectively). Open-source Llama3.1 shows competitive performance, particularly in entity recognition (Ent-F1: 0.942 on PII-Multi).

Entity type classification remains challenging. A

Method	GPT4o		Llama3.1		Qwen2.5		Llama3.1-SLM		Qwen2.5-SLM	
	F1	RougeL-F	F1	RougeL-F	F1	RougeL-F	F1	RougeL-F	F1	RougeL-F
Basic Method										
Naive	0.72	0.72	0.72	0.73	0.70	0.70	0.42	0.43	0.54	0.58
Advanced Method										
Self-CoT	0.76	0.77	0.75	0.75	0.73	0.73	0.53	0.54	0.54	0.58
Auto-CoT(3-shot)	0.75	0.75	**0.76**	0.77	**0.76**	0.76	**0.57**	0.58	0.54	0.58
Self-Consistency	**0.77**	0.77	0.71	0.72	0.71	0.72	0.49	0.50	0.49	0.53
PS-CoT	0.74	0.74	0.72	0.73	0.73	0.73	0.48	0.50	**0.56**	0.60
w/ Extra Information										
Naive w/ Choice	0.82	0.82	0.77	0.78	0.79	0.79	0.46	0.48	0.67	0.71

Table 4: Performance comparison on the Query-Unrelated PII Masking task (PII-single and PII-multi datasets). The best performance for each model (excluding Naive w/ Choice) is in **bold**.

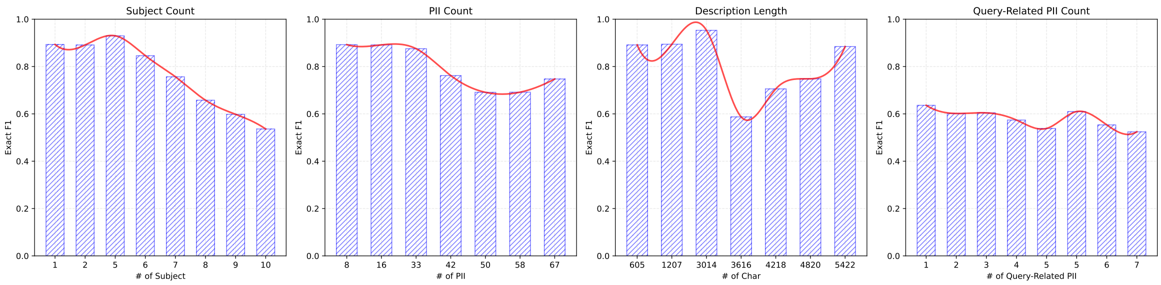

Figure 4: The performance of GPT-4o is correlated with the number of subject, the number of PII, decription length, and the number of query-related PII.

Method	Test-Hard		Test-Distract	
	F1	RougeL-F	F1	RougeL-F
Basic Method				
Naive	0.36	0.36	0.57	0.57
Advanced Method				
Self-CoT	0.45	0.45	0.66	0.67
Auto-CoT(3-shot)	0.45	0.40	0.62	0.63
Self-Consistency	**0.46**	0.46	0.62	0.63
PS-CoT	0.38	0.38	**0.67**	0.67
w/ Extra Information				
Naive w/ Choice	0.53	0.53	0.66	0.66

Table 5: Performance comparison on challenging test sets using GPT4o.

consistent gap between Strict-F1 and Ent-F1 scores indicates that accurate PII type classification poses greater challenges than entity boundary detection. This disparity becomes more pronounced in the PII-Distract dataset, suggesting increased difficulty in precise PII categorization under complex scenarios.

4.4 Performance on Query-Related PII Detection

Tables 7 presents the results on PII-single dataset across different model scales and prompting strategies.

Limited Performance of Current LLMs. State-of-the-art LLMs exhibit limited performance, with GPT4o achieving only 0.627 F1 score with naive prompting—substantially below human performance (0.951 F1). Open-source alternatives show competitive performance, with Qwen2.5 reaching 0.615 F1.

Impact of Advanced Prompting. Chain-of-thought approaches generally improve performance, with Self-Consistency and Auto-CoT proving most effective (0.716 F1 for GPT4o with Self-Consistency; 0.710 F1 for Qwen2.5 with Auto-CoT). However, these benefits are highly dependent on model scale—smaller models often show degraded performance with complex prompting strategies.

Effectiveness of Entity Candidates. Providing candidate PII entities (Naive w/ Choice) substantially improves performance across all models (e.g., GPT4o improves from 0.627 to 0.842 F1). However, practical applicability is limited as candidate entities are rarely available in real-world scenarios.

4.5 Privacy-Utility Tradeoff Analysis

We evaluated the tradeoff between privacy protection and query utility across different PII masking strategies

Experimental Setup. We selected 200 unique user descriptions from PII-Bench and applied three masking methods: (1) **No Mask**: Original text with all PII preserved; (2) **All PII Mask**: All de-

Baseline Models	PII-Single			PII-Multi			PII-Hard			PII-Distract		
	Strict-F1	Ent-F1	RougeL-F	Strict-F1	Ent-F1	RougeL-F	Strict-F1	Ent-F1	RougeL-F	Strict-F1	Ent-F1	RougeL-F
Traditional Model												
BiLSTM-CRF	-	0.851	-	-	0.828	-	-	0.684	-	-	0.787	-
API-based Large Language Model												
GPT4o	0.893	0.914	0.895	**0.891**	0.923	**0.893**	0.817	0.869	0.819	0.715	0.868	0.716
Claude3.5	0.858	0.891	0.862	0.890	0.920	0.892	0.813	0.857	0.818	**0.910**	**0.948**	**0.911**
DeepSeekV3	**0.903**	**0.921**	**0.905**	0.884	0.927	0.886	0.838	0.893	0.838	0.658	0.945	0.658
Open-source Large Language Model												
Llama3.1	0.881	0.913	0.883	0.883	**0.942**	0.884	**0.840**	**0.893**	**0.841**	0.834	0.946	0.835
Qwen2.5	0.866	0.908	0.869	0.853	0.918	0.855	0.804	0.876	0.806	0.647	0.941	0.649
Open-source Small Language Model												
Llama3.1-SLM	0.748	0.800	0.752	0.778	0.869	0.781	0.718	0.798	0.722	0.551	0.876	0.552
Qwen2.5-SLM	0.787	0.846	0.792	0.451	0.806	0.453	0.591	0.810	0.594	0.454	0.815	0.456

Table 6: Performance of baseline models under the PII Detection task. Results in **bold** indicate the best performance for each dataset and metric category.

Method	GPT4o		Llama3.1		Qwen2.5		Llama3.1-SLM		Qwen2.5-SLM	
	F1	RougeL-F	F1	RougeL-F	F1	RougeL-F	F1	RougeL-F	F1	RougeL-F
Basic Method										
Naive	0.63	0.63	0.63	0.63	0.62	0.62	0.33	0.33	0.41	0.41
Advanced Method										
Self-CoT	0.71	0.72	0.69	0.69	0.67	0.68	0.39	0.39	0.40	0.41
Auto-CoT(3-shot)	0.66	0.66	**0.70**	0.72	**0.71**	0.72	**0.43**	0.44	0.37	0.38
Self-Consistency	**0.72**	0.72	0.63	0.64	0.65	0.65	0.31	0.32	0.32	0.33
PS-CoT	0.65	0.65	0.65	0.66	0.67	0.67	0.35	0.36	**0.45**	0.46
w/ Extra Information										
Naive w/ Choice	0.84	0.84	0.76	0.76	0.83	0.83	0.52	0.52	0.77	0.77

Table 7: Performance comparison on the Query-Related PII Detection task (PII-single dataset).

Method	P	U	B
No Mask	0.00	1.00	0.50
All PII Mask	1.00	0.52	0.76
Query-unrelated PII Mask	0.83	0.89	0.86

Table 8: Privacy-utility tradeoff across different PII masking strategies. Balanced scores (B) combine both metrics with equal weights.

tected PII entities replaced with their corresponding tags; (3) **Query-unrelated PII Mask**: Only query-irrelevant PII entities masked. For each method, we generated responses using GPT4o (temperature=0, top_p=1).

Metrics. The privacy-utility tradeoff is evaluated through two metrics: **Privacy Score** (P): Quantifies the proportion of PII successfully protected in the processed text. **Utility Score** (U): Combines semantic similarity and LLM evaluation for quality assessment between mask and unmasked responses. Detailed computation procedures are provided in Appendix D.2.

Results. Table 8 shows that Query-unrelated PII Mask achieves optimal balance between privacy protection ($P = 0.83$) and utility preservation ($U = 0.89$), outperforming other strategies in balanced score ($B = 0.86$).

4.6 In-depth Performance Analysis

Factors Influencing Performance. Fig. 4 reveals several critical factors affecting model accuracy: performance degrades sharply beyond 5 subjects (F1 drops from 0.85 to 0.52), 33 PII entities, or 3000 characters in text length. Query-related entity count shows modest impact, with gradual decline from 1 to 7 entities.

Performance on Challenging Scenarios. Results on specialized test sets (Table 5) reveal significant performance degradation in complex scenarios. On Test-Hard, featuring high PII density and long texts, even the best-performing Self-Consistency approach achieves only 0.463 F1. Test-Distract's multi-subject scenarios pose similar challenges.

5 Conclusion

This paper introduces PII-Bench, a comprehensive evaluation framework comprising 2,842 test samples, along with a query-unrelated PII masking strategy. While current LLMs achieve strong performance in basic PII detection, they show limited capability in query-relevance assessment. Small-scale models demonstrate substantially lower performance across all tasks. These findings establish foundational benchmarks and exposes critical challenges in privacy-aware PII handling.

Limitations

Despite PII-Bench's contributions to privacy protection evaluation, several limitations merit acknowledgment. While the current dataset encompasses common privacy scenarios, it requires expansion into specialized domains such as medical records and financial transactions. Our automated synthesis methodology mitigates this limitation by enabling flexible dataset expansion across domains, languages, and cultural contexts, supporting continuous refinement of PII categories to meet evolving application requirements. The evaluation framework primarily assesses the accuracy of PII entity detection and query relevance determination, but lacks systematic evaluation of models' reasoning processes. Specifically, it does not fully capture how models interpret queries, derive information requirements, and make relevance judgments about sensitive information. This gap in assessment methodology limits our understanding of models' reasoning capabilities in real-world privacy protection scenarios.

Ethical Concerns

Throughout the development and implementation of PII-Bench, ethical considerations have remained our paramount priority. To ensure the evaluation dataset itself does not compromise privacy, we have implemented rigorous data synthesis and review protocols, with all sample data undergoing multiple rounds of scrutiny by professional security teams to guarantee the absence of real personal information. During the data generation process, we have carefully engineered our algorithms to ensure equitable representation across different demographic groups, establishing comprehensive human review mechanisms to verify that generated data remains free from bias and discriminatory content.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Anthropic. 2024. Claude 3.5 sonnet. `https://www.anthropic.com/news/claude-3-5-sonnet`.

Dimitris Asimopoulos, Ilias Siniosoglou, Vasileios Argyriou, Thomai Karamitsou, Eleftherios Fountoukidis, Sotirios K Goudos, Ioannis D Moscholios, Konstantinos E Psannis, and Panagiotis Sarigiannidis. 2024. Benchmarking advanced text anonymisation methods: A comparative study on novel and traditional approaches. *arXiv preprint arXiv:2404.14465*.

David Biesner, Rajkumar Ramamurthy, Robin Stenzel, Max Lübbering, Lars Hillebrand, Anna Ladi, Maren Pielka, Rüdiger Loitz, Christian Bauckhage, and Rafet Sifa. 2022. Anonymization of german financial documents using neural network-based language models with contextual word representations. *International Journal of Data Science and Analytics*, pages 1–11.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Tao Chen, Ruifeng Xu, Yulan He, and Xuan Wang. 2017. Improving sentiment analysis via sentence type classification using bilstm-crf and cnn. *Expert Systems with Applications*, 72:221–230.

Louise Deleger, Katalin Molnar, Guergana Savova, Fei Xia, Todd Lingren, Qi Li, Keith Marsolo, Anil Jegga, Megan Kaiser, Laura Stoutenborough, et al. 2013. Large-scale evaluation of automated clinical note de-identification and its impact on information extraction. *Journal of the American Medical Informatics Association*, 20(1):84–94.

Franck Dernoncourt, Ji Young Lee, Ozlem Uzuner, and Peter Szolovits. 2017. De-identification of patient notes with recurrent neural networks. *Journal of the American Medical Informatics Association*, 24(3):596–606.

Josep Domingo-Ferrer, David Sánchez, and Jordi Soria-Comas. 2022. *Database anonymization: privacy models, data utility, and microaggregation-based inter-model connections*. Springer Nature.

MM Douglass, GD Cliffford, Andrew Reisner, WJ Long, GB Moody, and RG Mark. 2005. De-identification algorithm for free-text nursing notes. In *Computers in Cardiology, 2005*, pages 331–334. IEEE.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Mark Elliot, Elaine Mackey, Kieron O'Hara, and Caroline Tudor. 2016. The anonymisation decision-making framework. *UKAN*.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Alistair EW Johnson, Lucas Bulgarelli, and Tom J Pollard. 2020. Deidentification of free-text medical records using pre-trained bidirectional transformers. In *Proceedings of the ACM Conference on Health, Inference, and Learning*, pages 214–221.

Qinbin Li, Junyuan Hong, Chulin Xie, Jeffrey Tan, Rachel Xin, Junyi Hou, Xavier Yin, Zhun Wang, Dan Hendrycks, Zhangyang Wang, et al. 2024. Llm-pbe: Assessing data privacy in large language models. *arXiv preprint arXiv:2408.12787*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Yang Liu, Yuanshun Yao, Jean-Francois Ton, Xiaoying Zhang, Ruocheng Guo Hao Cheng, Yegor Klochkov, Muhammad Faaiz Taufiq, and Hang Li. 2023. Trustworthy llms: A survey and guideline for evaluating large language models' alignment. *arXiv preprint arXiv:2308.05374*.

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.

Stephen Meisenbacher and Florian Matthes. 2024. Just rewrite it again: A post-processing method for enhanced semantic similarity and privacy preservation of differentially private rewritten text. *arXiv preprint arXiv:2405.19831*.

Multi-Linguality Multi-Functionality Multi-Granularity. 2024. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation.

Yuta Nakamura, Shouhei Hanaoka, Yukihiro Nomura, Naoto Hayashi, Osamu Abe, Shuntaro Yada, Shoko Wakamiya, and Eiji Aramaki. 2020. Kart: Privacy leakage framework of language models pre-trained with clinical records. *arXiv preprint arXiv:2101.00036*.

OpenAI. 2024. Hello gpt-4o. `https://openai.com/index/hello-gpt-4o`.

Anthi Papadopoulou, Yunhao Yu, Pierre Lison, and Lilja Øvrelid. 2022. Neural text sanitization with explicit measures of privacy risk. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 217–229.

Fatemeh Khoda Parast, Chandni Sindhav, Seema Nikam, Hadiseh Izadi Yekta, Kenneth B Kent, and Saqib Hakak. 2022. Cloud computing security: A survey of service-based models. *Computers & Security*, 114:102580.

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.

Patrick Ruch, Robert H Baud, Anne-Marie Rassinoux, Pierrette Bouillon, and Gilbert Robert. 2000. Medical document anonymization with a semantic lexicon. In *Proceedings of the AMIA Symposium*, page 729. American Medical Informatics Association.

Zhili Shen, Zihang Xi, Ying He, Wei Tong, Jingyu Hua, and Sheng Zhong. 2024. The fire thief is also the keeper: Balancing usability and privacy in prompts. *arXiv preprint arXiv:2406.14318*.

Shreya Singhal, Andres Felipe Zambrano, Maciej Pankiewicz, Xiner Liu, Chelsea Porter, and Ryan S Baker. 2024. De-identifying student personally identifying information with gpt-4. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 559–565.

Xiaofei Sun, Xiaoya Li, Jiwei Li, Fei Wu, Shangwei Guo, Tianwei Zhang, and Guoyin Wang. 2023. Text classification via large language models. *Preprint*, arXiv:2305.08377.

Xiongtao Sun, Gan Liu, Zhipeng He, Hui Li, and Xiaoguang Li. 2024. Deprompt: Desensitization and evaluation of personal identifiable information in large language model prompts. *arXiv preprint arXiv:2408.08930*.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. 2023. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. *arXiv preprint arXiv:2305.04091*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

IpKin Anthony Wong, Qi Lilith Lian, and Danni Sun. 2023. Autonomous travel decision-making: An early glimpse into chatgpt and generative ai. *Journal of Hospitality and Tourism Management*, 56:253–263.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024a. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Songhua Yang, Hanjie Zhao, Senbin Zhu, Guangyu Zhou, Hongfei Xu, Yuxiang Jia, and Hongying Zan. 2024b. Zhongjing: Enhancing the chinese medical capabilities of large language model through expert feedback and real-world multi-turn dialogue. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19368–19376.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*.

A Details about PII

A.1 PII Definition

In this section, we follow previous work by categorizing Personally Identifiable Information (PII) into the following two categories(Elliot et al., 2016,Domingo-Ferrer et al., 2022,Papadopoulou et al., 2022):

- *Direct identifiers*: Information that can uniquely identify an individual within a dataset(e.g. name, social security number, email address, etc).
- *Quasi identifiers*: Information that cannot uniquely identify an individual on their own but can do so when combined with other quasi-identifiers(e.g. age, gender, occupation, etc.

Because of their high sensitivity or the potential to indirectly identify an individual, both direct and quasi-identifiers are governed by strict legal and privacy standards to ensure personal privacy.

A.2 PII Types

Unlike the PII types presented by Papadopoulou et al.'s (2022), our classification does not include the MISC category. This exclusion is due to the ambiguous definition of the MISC category and its unclear boundaries with other categories.

The definitions of the seven categories are as follows:

PER: Refers to individuals' names, including full names, aliases, and social media usernames.

CODE: Encompasses identifying numbers and codes like social security numbers, phone numbers, passport numbers, email addresses, etc.

LOC: Covers geographical locations such as home or work addresses, cities, countries, etc.

ORG: Pertains to the names of entities like companies, schools, public institutions, etc.

DEM: Represents demographic information including age, gender, nationality, occupation, education level, etc.

DATETIME: Indicates specific dates, times, or durations, such as birthdates, appointment times, etc.

QUANTITY: Refers to significant numerical data like monthly income, expenditures, loan amount, credit score, etc.

A.3 Statistics of PII Types

Figure 5 and Table 9 present the distribution of PII types across our datasets: PII-single (1,214 samples), PII-multi (1,228 samples), PII-hard (200 samples), and PII-distract (200 samples).

- **Type Frequencies**: Organization (ORG) and Code-based identifiers (CODE) constitute significant portions across all datasets, with 17.09% and 15.74% in PII-single, and 13.47% and 15.31% in PII-multi, respectively. This distribution reflects the prevalence of institutional affiliations and digital identifiers in real-world scenarios.
- **Dataset Composition**: PII-multi contains 16,136 PII entities across all categories, maintaining balanced proportions ranging from 13.47% to 15.77% for most types. PII-single follows a similar pattern with 9,303 entities, demonstrating consistent coverage across different PII categories.
- **Specialized Test Sets**: PII-distract, despite comprising only 200 samples, contains 10,211 PII entities due to its multi-description design. PII-hard maintains balanced type coverage with 1,834 entities, with proportions varying from 12.10% to 16.58%.

B PII Entity Generation Methods

The generation of PII entities requires careful consideration of both structural constraints and semantic plausibility. We employ two complementary approaches for entity generation: rule-based generation for structured PII types and language model-based generation for context-dependent information.

B.1 Rule-based Generation

For PII types with well-defined formats or enumerable value sets, we implement deterministic generation methods. These methods encompass

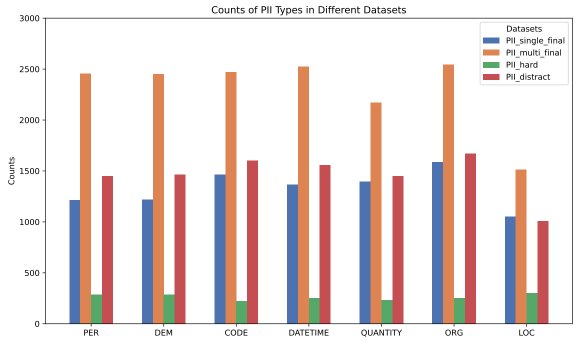

Figure 5: Distribution of PII types across different datasets in PII-Bench.

both custom rule-based algorithms and the Faker library's standardized functions. The rule-based approach is particularly effective for:

1. Identification Numbers: Generating valid formats for social security numbers, passport numbers, and employee IDs while maintaining regional compliance.

2. Contact Information: Creating syntactically correct email addresses, phone numbers, and IP addresses.

3. Financial Data: Producing properly formatted credit card numbers, bank account numbers, and other numerical identifiers with appropriate check digits.

4. Temporal Information: Generating dates, times, and durations within reasonable ranges and formats.

Type	PII_single		PII_multi		PII_hard		PII_distract	
	#	%	#	%	#	%	#	%
PER	1,214	13.05	2,456	15.22	286	15.59	1,449	14.19
DEM	1,220	13.12	2,450	15.18	286	15.59	1,467	14.37
CODE	1,464	15.74	2,470	15.31	222	12.10	1,605	15.72
ORG	1,590	17.09	2,544	13.47	251	13.69	1,673	16.38
LOC	1,053	11.32	1,516	15.77	304	16.58	1,008	9.87
DATETIME	1,368	14.70	2,526	15.65	251	13.69	1,559	15.27
QUANTITY	1,394	14.98	2,174	9.40	234	12.76	1,450	14.20
Total	9,303	100	16,136	100	1,834	100	10,211	100

Table 9: Detailed statistics of PII types across datasets. For each dataset, we report both the absolute count (#) and relative percentage (%) of each PII type.

B.2 Language Model-based Generation

For PII types requiring contextual understanding and real-world knowledge, we leverage large language models through carefully designed prompts. This approach is essential for generating:

1. Location Information: Coherent and geographically accurate addresses, landmarks, and regional descriptions.

2. Organizational Entities: Plausible names for educational institutions, companies, and other organizations that reflect real-world naming conventions.

3. Demographic Attributes: Culturally appropriate and consistent demographic information, including ethnicity, nationality, and educational background.

B.3 Entity Categories and Generation Methods

Table 14 presents a comprehensive mapping of PII types to their respective generation methods. The table systematically categorizes 55 distinct PII entities across seven main categories: Personal Identifiers (PER), Codes and Numbers (CODE), Location Information (LOC), Organizational Affiliations (ORG), Demographic Information (DEM), Temporal Data (DATETIME), and Quantitative Values (QUANTITY).

C Human Evaluation Details

The human evaluation of PII-Bench was conducted with 25 graduate students specializing in data security and privacy protection. All evaluators were pursuing their Master's or Ph.D. degrees with at least two years of research experience in privacy-preserving machine learning or data protection systems. The evaluation process consisted of three phases: preparation, evaluation, and validation.

During the preparation phase, participants attended a 4-hour training session covering PII taxonomy, recognition guidelines, and query-related detection criteria. The session included hands-on practice with representative cases from each dataset component. Participants then completed a qualification test featuring 20 diverse instances, requiring 90% agreement with expert assessments to proceed to the formal evaluation.

During the evaluation phase, participants used our specialized platform designed for systematic PII assessment. To maintain consistent performance, we limited evaluation sessions to two hours and distributed instances across a two-week period. The platform automatically tracked assessment time and accuracy metrics while enforcing our evaluation protocol: participants first performed PII detection by marking entity spans, linking them to subjects, and assigning PII types, before proceeding to query-related detection.

Our validation process incorporated both automated and manual checks to ensure assessment quality. The platform automatically verified assessment completeness and format consistency. Cases with substantial disagreement (Fleiss' kappa < 0.6) underwent expert review by two authors with extensive experience in privacy-preserving systems. Evaluators received detailed feedback on their performance and participated in discussion sessions to resolve systematic discrepancies.

Compensation was structured to encourage both accuracy and efficiency, with a base rate of \$30 per hour and performance bonuses based on agreement with other evaluators.

D Experiments Details

D.1 Evaluation Metrics

The evaluation framework employs distinct metrics for PII detection and query-related detection tasks.

For PII detection, let $\mathcal{P} = \{p_1, ..., p_m\}$ denote the predicted subject set and $\mathcal{G} = \{g_1, ..., g_n\}$ denote the ground truth subject set. Each subject p_i or g_j contains a set of entity-type pairs $\{(e, t)\}$, where e represents the entity span and t represents its PII type.

For each subject pair (p_i, g_j), we compute three types of evaluation metrics:

1. Strict Matching: Both entity spans and their types must match exactly:

$$P_{strict}(p_i, g_j) = \frac{|E_{p_i} \cap E_{g_j}|}{|E_{p_i}|} \tag{1}$$

$$R_{strict}(p_i, g_j) = \frac{|E_{p_i} \cap E_{g_j}|}{|E_{g_j}|} \tag{2}$$

$$F1_{strict}(p_i, g_j) = \frac{2 \cdot P_{strict}(p_i, g_j) \cdot R_{strict}(p_i, g_j)}{P_{strict}(p_i, g_j) + R_{strict}(p_i, g_j)} \tag{3}$$

where E_{p_i} and E_{g_j} are the sets of entity-type pairs.

2. Entity-only Matching: Only entity spans need to match:

$$P_{ent}(p_i, g_j) = \frac{|S_{p_i} \cap S_{g_j}|}{|S_{p_i}|} \tag{4}$$

$$R_{ent}(p_i, g_j) = \frac{|S_{p_i} \cap S_{g_j}|}{|S_{g_j}|} \tag{5}$$

$$F1_{ent}(p_i, g_j) = \frac{2 \cdot P_{ent}(p_i, g_j) \cdot R_{ent}(p_i, g_j)}{P_{ent}(p_i, g_j) + R_{ent}(p_i, g_j)} \tag{6}$$

where S_{p_i} and S_{g_j} are the sets of entity spans.

The optimal subject matching M^* is determined by maximizing the strict F1 score:

$$M^* = \max_{M \in \mathcal{M}} \sum_{(p_i, g_j) \in M} F1_{strict}(p_i, g_j) \tag{7}$$

where $\mathcal{M}$ denotes all possible one-to-one mappings between predicted and ground truth subjects.

The final recognition scores are computed over the optimal matching pairs:

$$P_{strict} = \frac{1}{|\mathcal{P}|} \sum_{(p_i, g_j) \in M^*} P_{strict}(p_i, g_j) \tag{8}$$

$$R_{strict} = \frac{1}{|\mathcal{G}|} \sum_{(p_i, g_j) \in M^*} R_{strict}(p_i, g_j) \tag{9}$$

$$F1_{strict} = \frac{1}{max(\mathcal{P}, \mathcal{G})} \sum_{(p_i, g_j) \in M^*} F1_{strict}(p_i, g_j) \tag{10}$$

P_{span}, R_{span}, and $F1_{span}$ are computed analogously.

For query-related detection, given a predicted entity set $\mathcal{E}_p$ and ground truth set $\mathcal{E}_g$, we compute:

$$P_{query} = \frac{|\mathcal{E}_p \cap \mathcal{E}_g|}{|\mathcal{E}_p|} \tag{11}$$

$$R_{query} = \frac{|\mathcal{E}_p \cap \mathcal{E}_g|}{|\mathcal{E}_g|} \tag{12}$$

$$F1_{query} = \frac{2 \cdot P_{query} \cdot R_{query}}{P_{query} + R_{query}} \tag{13}$$

For both PII detection and query-related detection tasks, we additionally employ Rouge-L based fuzzy matching to handle partial matches between entity spans. Instead of using exact set intersection, the Rouge-L score is used to measure textual similarity between entities:

$$P_{fuzzy} = \frac{1}{|\mathcal{E}_p|} \sum_{e_p \in \mathcal{E}_p} \max_{e_g \in \mathcal{E}_g} Rouge\text{-}L(e_p, e_g) \tag{14}$$

$$R_{fuzzy} = \frac{1}{|\mathcal{E}_g|} \sum_{e_g \in \mathcal{E}_g} \max_{e_p \in \mathcal{E}_p} Rouge\text{-}L(e_p, e_g) \tag{15}$$

$$F1_{fuzzy} = \frac{2 \cdot P_{fuzzy} \cdot R_{fuzzy}}{P_{fuzzy} + R_{fuzzy}} \tag{16}$$

where $Rouge\text{-}L(e_p, e_g)$ computes the longest common subsequence-based F-score between predicted entity e_p and ground truth entity e_g.

D.2 Privacy-Utility Tradeoff Metrics

We develop a comprehensive evaluation framework to quantify the tradeoff between privacy protection and query utility across different PII masking strategies.

D.2.1 Privacy Protection Metric

Let $\mathcal{E} = \{e_1, ..., e_n\}$ denote the set of PII entities in the original text T_o, and T_m represent the masked text. The privacy score P measures the proportion of PII entities successfully protected:

$$P = 1 - \frac{\sum_{e \in \mathcal{E}} C(e, T_m)}{\sum_{e \in \mathcal{E}} C(e, T_o)} \tag{17}$$

where $C(e, T)$ counts occurrences of entity e in text T. A score of $P = 1$ indicates complete protection, while $P = 0$ indicates no protection.

D.2.2 Utility Preservation Metrics

To measure utility preservation, we employ two complementary approaches:

Semantic Similarity We compute embedding-based similarity between responses generated from masked prompts (R_m) and unmasked prompts (R_o):

$$U_s = \cos(\mathbf{v}_{R_m}, \mathbf{v}_{R_o}) \tag{18}$$

where $\mathbf{v}_R$ is the text embedding generated by BGE-M3 (Multi-Granularity, 2024), and $\cos(\cdot, \cdot)$ computes cosine similarity. This metric captures semantic preservation independent of exact wording.

LLM-as-Judge Evaluation We employ Claude-3.7-Sonnet to assess response quality through direct comparison:

$$U_l = Judge(R_m^1, R_m^2, R_o) \tag{19}$$

where R_m^1 and R_m^2 represent responses from two different masking strategies, and R_o is the reference response from unmasked text. The judge assigns numerical ratings $r \in [1, 10]$ to each masked response based on how well it preserves the query intent compared to the reference. To mitigate position bias, we randomly alternate the presentation order of responses.

D.2.3 Balanced Metric

To quantify the overall effectiveness of masking strategies, we compute a balanced score B that combines privacy protection and utility preservation:

$$B = \alpha P + (1 - \alpha) U \tag{20}$$

where $\alpha \in [0, 1]$ is a weighting parameter that determines the relative importance of privacy versus utility. In our experiments, we use $\alpha = 0.5$ to assign equal importance to both aspects.

The complete prompt template for LLM-as-Judge evaluation is provided in Figure 18.

	Qwen2.5-0.5B		Qwen2.5-1.5B		Qwen2.5-3B	
Method	**F1**	**RougeL-F**	**F1**	**RougeL-F**	**F1**	**RougeL-F**
Basic Method						
Naive	0.0041	0.0041	0.2131	0.2185	0.1691	0.1699
Advanced Method						
Self-CoT	0.009	0.0095	0.2002	0.2051	0.2884	0.2909
Auto-CoT(3-shot)	0.0258	0.0268	0.2383	0.2466	0.381	0.384
Self-Consistency	0.0018	0.0018	0.1057	0.1088	0.2694	0.2774
PS-CoT	0.0088	0.009	0.1887	0.1959	0.293	0.2956
w/ PII Detection						
Naive w/ Choice	0.3978	0.3986	0.4357	0.4357	0.542	0.5437

Table 10: Performance comparison on the Query-Related PII Detection task (PII-single dataset).

	Qwen2.5-0.5B		Qwen2.5-1.5B		Qwen2.5-3B	
Method	**F1**	**RougeL-F**	**F1**	**RougeL-F**	**F1**	**RougeL-F**
Basic Method w/ PII Detection						
Naive	0.0016	0.0025	0.0912	0.1018	0.3257	0.3321
Advanced Method w/ PII Detection						
Self-CoT	0.0016	0.0029	0.0846	0.0945	0.3764	0.3841
Auto-CoT(3-shot)	0.0019	0.0032	0.0927	0.1038	0.4247	0.4327
Self-Consistency	0.0016	0.0029	0.0907	0.1020	0.3414	0.3486
PS-CoT	0.0016	0.0025	0.0920	0.1033	0.3765	0.3840
w/ PII Detection						
Naive w/ Choice	0.0011	0.0017	0.0722	0.0800	0.3918	0.3994

Table 11: Performance comparison on the Query-Unrelated PII Masking task (PII-single and PII-multi datasets).

D.3 Additional Experiments with Smaller Language Models

We have conducted a comprehensive evaluation on smaller, deployment-ready models (0.5B-3B parameters) to assess their viability for on-device PII protection.

As shown in Tables 10, 11, and 12, our results reveal a significant performance gap between deployment-ready models and larger proprietary systems. The 0.5B and 1.5B models exhibit extremely limited capabilities across all tasks (with F1 scores generally below 0.2), rendering them practically unusable for real-world PII protection. While the 3B model shows modest potential (F1 scores approaching 0.6 on simpler datasets), its performance remains substantially inferior to larger

	PII-Single			PII-Multi			PII-Hard			PII-Distract		
Baseline Models	**Strict-F1**	**Ent-F1**	**RougeL-F**	**Strict-F1**	**Ent-F1**	**RougeL-F**	**Strict-F1**	**Ent-F1**	**RougeL-F**	**Strict-F1**	**Ent-F1**	**RougeL-F**
Qwen2.5-0.5B	0.002	0.0029	0.002	0.0013	0.0042	0.0013	0.0034	0.0042	0.0034	0.009	0.0207	0.009
Qwen2.5-1.5B	0.1846	0.2069	0.1865	0.1057	0.1794	0.1071	0.1474	0.1976	0.1502	0.0728	0.1549	0.0733
Qwen2.5-3B	0.5929	0.6289	0.5962	0.6189	0.7067	0.6212	0.5202	0.5921	0.5237	0.3717	0.6293	0.3731

Table 12: Performance of baseline models under the PII Detection task.

models, particularly on challenging scenarios (e.g., PII-Hard, PII-Distract). These findings underscore the tension between privacy goals and model capabilities: while smaller on-device models would be preferable from a privacy perspective, they currently lack the sophistication needed for reliable PII management.

D.4 Additional Results

Table 13 compares different prompting strategies on PII-multi dataset.

D.5 Prompt Details

This section presents the prompts used throughout our experiments. For the PII detection task, we employ the template shown in Figure 11. For query-related PII detection, we design and evaluate six distinct prompting strategies. Figure 12 displays the **Naive** prompts, Figure 13 presents the **Naive w/ Choice** prompts, Figure 14 features the **Self-CoT** prompts, Figure 15 reveals the **Auto-CoT** prompts, Figure 16 exhibits the **Self-Consistency** prompts,and Figure 17 displays the **PS-CoT** prompts.

E PII Annotation System

We developed a specialized web-based annotation platform to facilitate the systematic evaluation of PII detection and query-related detection capabilities. The platform implements a two-stage annotation process, ensuring comprehensive coverage of both fundamental PII entity identification and contextual relevance assessment.

E.1 PII Detection Interface

As shown in Figure 6, the PII detection interface enables annotators to identify and categorize PII entities within user descriptions. The interface provides the following key functionalities:

- Entity Detection: Annotators can highlight text spans containing PII entities directly in the user description.
- Type Classification: Each identified entity is assigned a specific PII type (e.g., PER for person names, ORG for organizations, LOC for locations).
- Subject Association: Entities are linked to their corresponding subjects using alphabetical identifiers (e.g., A, B) to maintain relationship clarity in multi-subject scenarios.
- Span Verification: The interface displays start and end positions for each entity span, ensuring precise boundary detection.

E.2 Query-Related Detection Interface

Figure 7 illustrates the interface for query-related PII detection, which builds upon the recognition results to assess contextual relevance:

- Query Context: The interface presents both the user description and the associated query, providing complete context for relevance assessment.
- Entity Selection: Annotators identify PII entities crucial for addressing the query, with the interface highlighting pre-identified entities from the recognition phase.
- Subject Verification: For selected query-related entities, annotators must verify the subject associations to ensure consistency across tasks.
- Relevance Validation: The interface includes a review mechanism to confirm that selected entities are both necessary and sufficient for query resolution.

E.3 Query-unrelated PII Masking Visualization

To validate the effectiveness of privacy protection while maintaining query relevance, we implemented a masking visualization interface (Figure 8):

- Original Context: Displays the complete user description with all PII entities highlighted.

Method	GPT4o		Llama3.1		Qwen2.5		Llama3.1-SLM		Qwen2.5-SLM	
	F1	RougeL-F	F1	RougeL-F	F1	RougeL-F	F1	RougeL-F	F1	RougeL-F
Basic Method										
Naive	0.600	0.602	0.611	0.614	0.596	0.603	0.240	0.333	0.405	0.413
Advanced Method										
Self-CoT	0.675	0.681	0.638	0.643	0.626	0.632	0.354	0.362	0.392	0.397
Auto-CoT(3-shot)	0.629	0.640	**0.650**	0.662	**0.657**	0.665	**0.393**	0.402	0.391	0.394
Self-Consistency	**0.685**	0.692	0.602	0.605	0.614	0.620	0.263	0.269	0.288	0.293
PS-CoT	0.618	0.620	0.624	0.631	0.636	0.643	0.291	0.300	**0.431**	0.436
w/ Extra Information										
Naive w/ Choice	0.846	0.846	0.775	0.775	0.804	0.804	0.387	0.388	0.743	0.743

Table 13: Performance comparison on the Query-Related PII Detection task (PII-multi dataset).

- Masked View: Shows the description with non-relevant PII entities replaced by their corresponding type tags (e.g., <Nickname>, <Phone Number>).
- Key Information Display: Preserves query-related PII entities while maintaining readability and semantic coherence.

E.4 Annotation Guidelines and Quality Control

To ensure annotation consistency and quality, we established comprehensive guidelines and implemented several control measures:

- Entity Span Guidelines: Annotators must select the minimal text span that completely captures the PII entity while maintaining semantic integrity.
- Inter-annotator Agreement: Each sample is independently annotated by multiple annotators, with disagreements resolved through majority voting or expert review.
- Validation Checks: The platform implements automatic validation rules to detect potential inconsistencies or missing annotations.
- Iterative Refinement: Regular review sessions are conducted to discuss challenging cases and update guidelines based on annotator feedback.

For quality assurance, we randomly sampled 10% of the annotations for expert review, achieving an inter-annotator agreement of 95.1% for PII detection and 91.5% for query-related detection across all annotators.

Not Finished

PII Type	Entity Category	Generation Approach	Format Constraints
PER	Full Name	Rule-based	[First Name] [Last Name]
	Social Media Handle	Rule-based	[@][a-zA-Z0-9]5,15
	Nickname	LLM-based	-
CODE	Social Security Number	Rule-based	XXX-XX-XXXX
	Driver's License	Rule-based	[A-Z][0-9]8
	Bank Account	Rule-based	[0-9]10,12
	Credit Card	Rule-based	[0-9]16
	Phone Number	Rule-based	+[0-9]1,3-[0-9]10
	IP Address	Rule-based	IPv4/IPv6 format
	Email Address	Rule-based	[user]@[domain].[tld]
	Password Hash	Rule-based	SHA-256
	Passport Number	Rule-based	[A-Z][0-9]8
	Tax ID	Rule-based	[0-9]9
	Employee ID	Rule-based	[A-Z]2[0-9]6
	Student ID	Rule-based	[0-9]8
LOC	Street Address	LLM-based	-
	City/Region	LLM-based	-
	Landmark	LLM-based	-
ORG	Company Name	LLM-based	-
	Educational Institution	LLM-based	-
	Government Agency	LLM-based	-
	NGO	LLM-based	-
	Healthcare Facility	LLM-based	-
DEM	Occupation	Rule-based	Predefined list
	Age	Rule-based	[0-9]1,3
	Gender	Rule-based	Binary/Non-binary
	Height	Rule-based	[0-9]3cm/[0-9]'[0-9]"
	Weight	Rule-based	[0-9]2,3kg/lbs
	Blood Type	Rule-based	A/B/O[+-]
	Sexual Orientation	Rule-based	Predefined list
	Nationality	LLM-based	-
	Ethnicity	LLM-based	-
	Race	LLM-based	-
	Religious Belief	LLM-based	-
	Political Affiliation	LLM-based	-
	Education Level	LLM-based	-
	Academic Degree	LLM-based	-
	Physical Features	LLM-based	-
	Medical Condition	LLM-based	-
	Disability Status	LLM-based	-
DATETIME	Date	Rule-based	YYYY-MM-DD
	Time	Rule-based	HH:MM:SS
	Duration	Rule-based	[0-9]+[dhms]
QUANTITY	Monthly Income	Rule-based	[Currency][0-9]+
	Monthly Expenses	Rule-based	[Currency][0-9]+
	Account Balance	Rule-based	[Currency][0-9]+
	Loan Amount	Rule-based	[Currency][0-9]+
	Annual Bonus	Rule-based	[Currency][0-9]+
	Credit Limit	Rule-based	[Currency][0-9]+
	Social Security Payment	Rule-based	[Currency][0-9]+
	Tax Payment	Rule-based	[Currency][0-9]+
	Debt Ratio	Rule-based	[0-9]1,2.[0-9]2%
	Investment Return	Rule-based	[0-9]1,2.[0-9]2%
	ROI	Rule-based	[0-9]1,2.[0-9]2%
	Credit Score	Rule-based	[300-850]

Table 14: Comprehensive categorization of PII entities and their generation methods. Rule-based generation follows specific format constraints, while LLM-based generation produces contextually appropriate content without rigid formatting requirements.

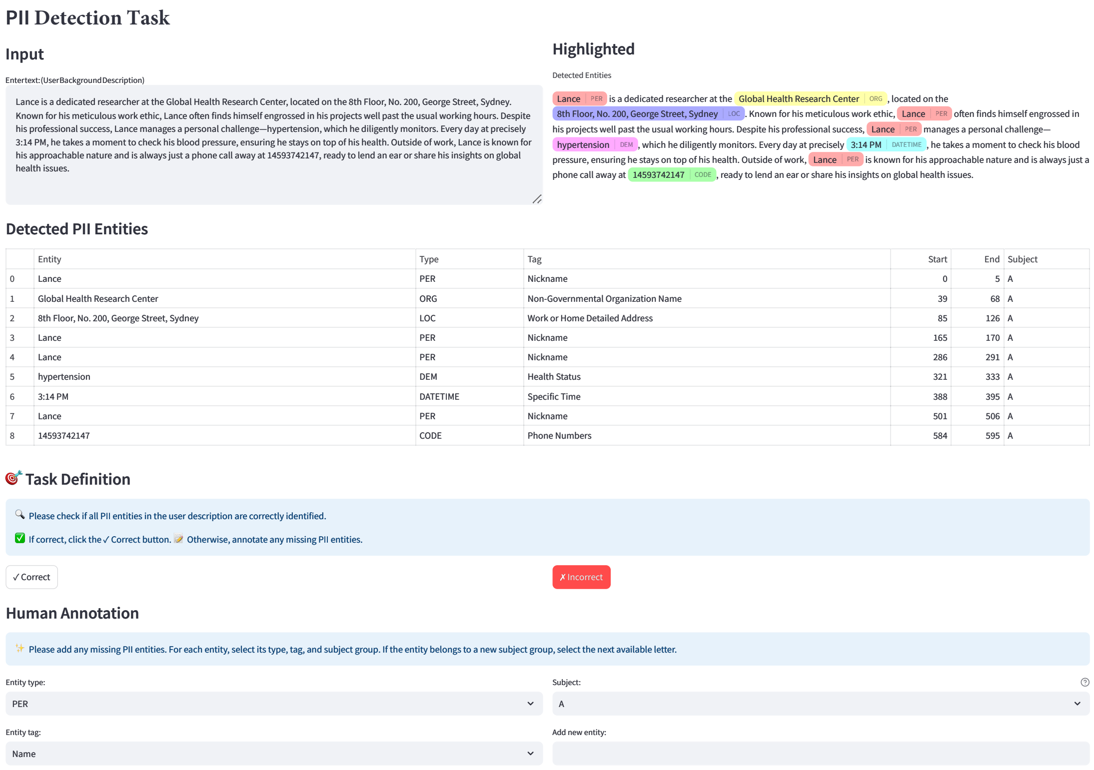

Figure 6: Web Demo for the PII Detection Task

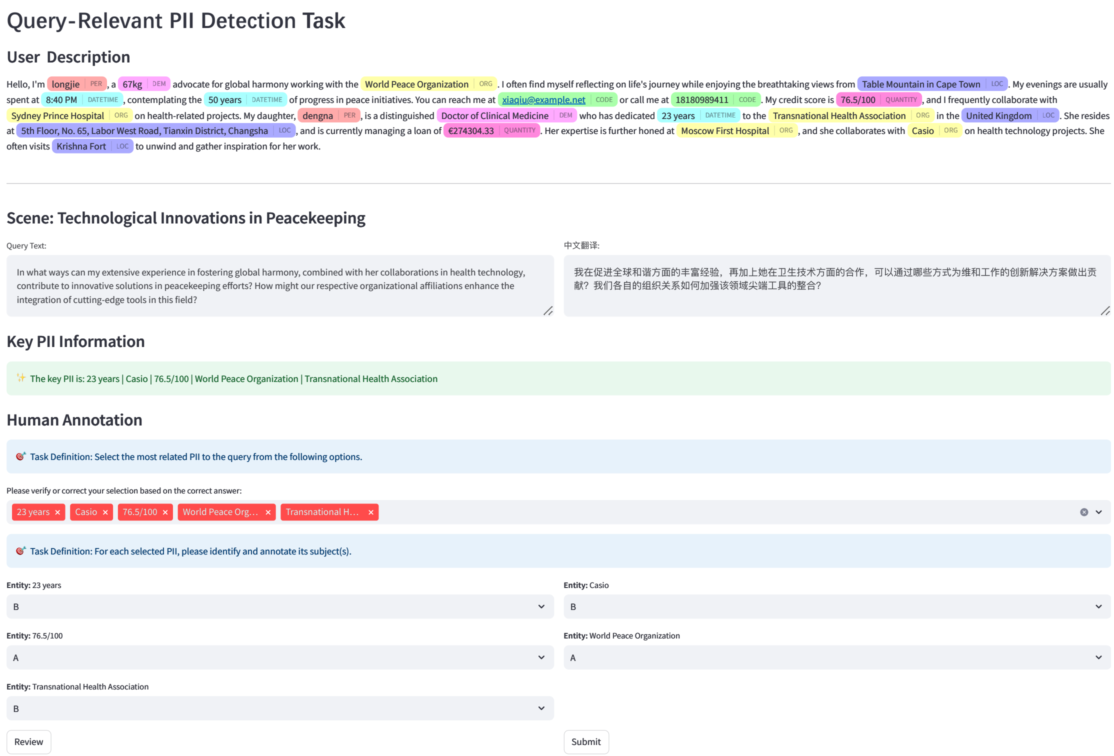

Figure 7: Web Demo for the Query-Related PII Detection Task

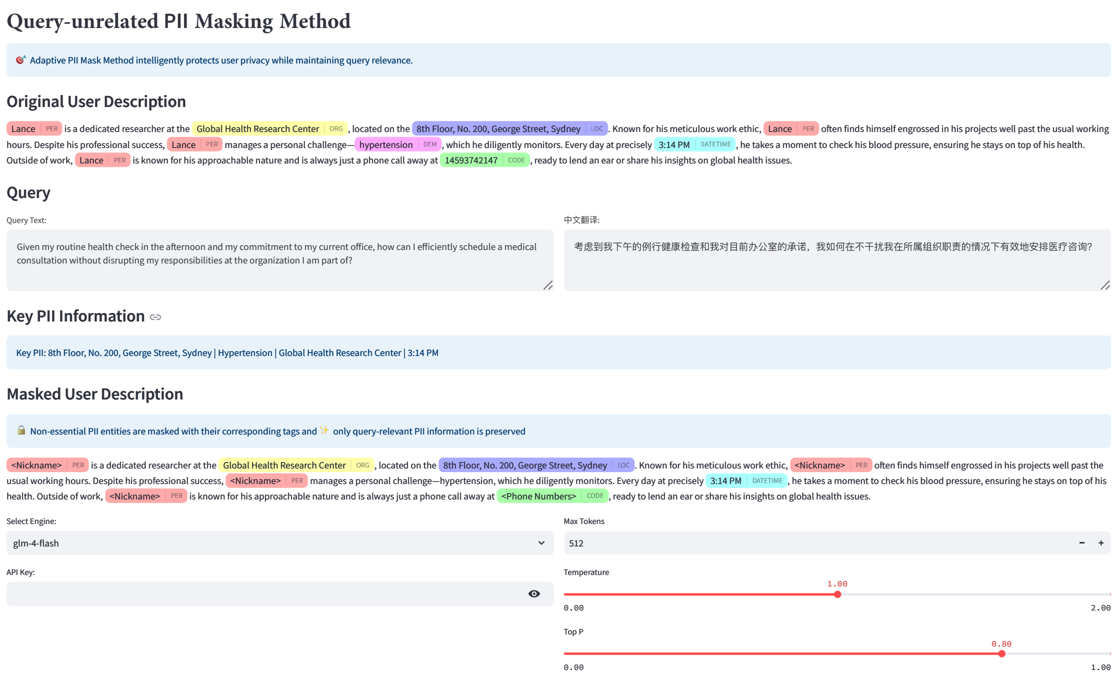

Figure 8: Web Demo for the Query-unrelated PII Masking Method

Consistency Optimization Prompt of Single Subject

You are a character feature selector tasked with identifying and refining logically consistent feature combinations.I will provide you with character features. Your role is to identify any features that have obvious logical conflicts or inconsistencies, and modify them to create a coherent set while preserving their core classifications.

Requirements:
1. The selected character features must be logically consistent with real-world expectations, with no obvious conflicts.
2. When resolving conflicts, modify only the feature entities while keeping their PII types and classifications unchanged.
3. Modified feature entities must remain within the same PII type and classification categories as their originals.
4. Aim to maintain as many features as possible, ideally matching the original count or coming as close as feasible.

Character Features
<PII Type> <Entity Category> <PII Entity>
{usr_features}

Please provide your output in the following format:
- Under "## Reason:", explain your selection and modification process
- Under "## Final Features:", list the final selected features as JSON objects in the format {{"label": xxx, "tag": yyy, "entity": zzz}} with no additional content or line breaks where xxx is the PII type, yyy is the entity category, and zzz is the PII entity.

Reason: [Explain your selection and modification process]
Final Features: [{{"label": xxx, "tag": yyy, "entity": zzz}}, {{"label":xxx, "tag": yyy, "entity": zzz}}, ...]"""

Figure 9: Prompt of Consistency Optimization for Single-Subject

Consistency Optimization Prompt of Multi Subject

You are a character feature selector tasked with identifying and refining logically consistent feature combinations.I will provide you with character features for different subjects and their relationships. Your role is to:
1. Identify any logical conflicts or inconsistencies between features
2. Modify conflicting features while maintaining their PII types and categories
3. Ensure all features align with the given relationship between subjects

Requirements:
1. Selected features must be logically consistent and align with the relationship between subjects
2. For relationships:
- "Intersection" can indicate friends or colleagues
- "Contains" can indicate parent-child relationships
- "No Intersection" indicates strangers
3. When modifying conflicting features:
- Maintain the original PII type and category
- Only modify the entity value
- New entity must belong to the same category
4. Maximize the number of selected features:
- Aim to keep the original count
- If not possible, get as close as possible

Subject A Features
<PII Type> <Entity Category> <PII Entity>
{usr_features_a}

Subject B Features
<PII Type> <Entity Category> <PII Entity>
{usr_features_b}

Relationship Between Subjects
{rel}

Please provide your output in the following format:
- Under "## Reason:", explain your selection and modification process
- Under "## Final Features A:" or "## Final Features B:", list the final selected features as JSON objects in the format {{"label": xxx, "tag": yyy, "entity": zzz}} with no additional content or line breaks where xxx is the PII type, yyy is the entity category, and zzz is the PII entity.

Reason: [Explain your selection and modification process]
Final Features A: [{{"label":xxx, "tag": yyy, "entity": zzz}}, {{"label":xxx, "tag": yyy, "entity": zzz}}, ...]
Final Features B: [{{"label":xxx, "tag": yyy, "entity": zzz}}, {{"label":xxx, "tag": yyy, "entity": zzz}}, ...]

Figure 10: Prompt of Consistency Optimization for Multi-Subject

PII Detection Prompt

Please identify the PII entities and their corresponding PII types for each distinct individual mentioned in the conversation transcript, including both speakers and referenced individuals.
The PII types are defined as follows:
{pii_definition}
PII types include: ["PER","CODE","LOC","ORG","DEM","DATETIME","QUANTITY"]

Task Description:
Your task is to:
1. Identify ALL distinct individuals mentioned in the text, including:
- Primary speakers (marked with [PER_X])
- Individuals mentioned within others' statements
- Referenced colleagues, family members, or associates

2. For each identified individual, extract their associated PII entities, ensuring:
- Each entity is in its smallest viable text span
- Entities are correctly categorized by type
- Cross-referenced information is attributed to the correct individual

Important Rules:
1. Treat each individual as a separate subject, even if mentioned within another person's statement
2. Include both explicitly named individuals and those referenced through relationships
3. Maintain clear boundaries between different individuals' information
4. Extract exact entity spans without additional context
5. Preserve special characters in codes and quantities
6. Handle both direct mentions and indirect references

Given conversation transcript:
{user_desc}
Required Output Format:
For each identified individual (both speakers and mentioned persons), output:
Subject {{N}} {{ent1: type1, ent2: type2, ...}}
Example:
Input text: "[PER_1]: I'm Alex, working at Google. My friend Bob, who is 25 years old, works at Apple."
Expected output:
Subject {{1}} {{"Alex": "PER", "Google": "ORG"}}
Subject {{2}} {{"Bob": "PER", "25 years": "DATETIME", "Apple": "ORG"}}

Begin analysis now:

Figure 11: Prompt used for the PII Detection task

Basic Query-related PII Detection Prompt

Please identify highly relevant PII (Personally Identifiable Information) entities from the background description PII entities that directly address or relate to the user's query.

Rules:
- Extract entities in their smallest possible span
- Exclude all person names
- Focus only on entities crucial for answering the query
- Return entities exactly as they appear in the text

Background description:
{desc}
Query:
{query}

Your output will contain the following format:
Answer: List the relevant PII entities, each enclosed in double quotes (""). Return only the list without explanation. Example: ["key_pii_1", ..., "key_pii_n"]

Please have your output follow the format below: (if there is only one entity, please output ["key_pii_1"]):
Answer: ["key_pii_1", ..., "key_pii_n"]

Figure 12: Prompt of Naive Method

Choice-Based Query-related PII Detection Prompt

From the following options, Please identify highly relevant PII (Personally Identifiable Information) entities from the background description PII entities that directly address or relate to the user's query.

Rules:
- Extract entities in their smallest possible span
- Exclude all person names
- Focus only on entities crucial for answering the query
- Return entities exactly as they appear in the text
- Select only from the provided options

Background description:
{desc}
Query:
{query}
Options:
{choices}

Your output will contain the following format:
Answer: List the relevant PII entities, each enclosed in double quotes (""). Return only the list without explanation. Example: ["key_pii_1", ..., "key_pii_n"]

Please have your output follow the format below: (if there is only one entity, please output ["key_pii_1"]):
Answer: ["key_pii_1", ..., "key_pii_n"]

Figure 13: Prompt of Naive /w Choice Method

Chain-of-Thought Query-related PII Detection Prompt

Please identify highly relevant PII (Personally Identifiable Information) entities from the background description PII entities that directly address or relate to the user's query.

Rules:
- Extract entities in their smallest possible span
- Exclude all person names
- Focus only on entities crucial for answering the query
- Return entities exactly as they appear in the text

Background description:
{desc}
Query:
{query}

Your output will contain the following format:
Thought: Explain your reasoning step by step for selecting the relevant PII entities.
Answer: List the relevant PII entities, each enclosed in double quotes (""). Return only the list without explanation. Example: ["key_pii_1", ..., "key_pii_n"]

Please have your output follow the format below: (if there is only one entity, please output ["key_pii_1"]):
Thought: xxx
Answer: ["key_pii_1", ..., "key_pii_n"]

Figure 14: Prompt of Self-CoT Method

Auto Chain-of-Thought Query-related PII Detection Prompt with Examples

Please identify highly relevant PII (Personally Identifiable Information) entities from the background description PII entities that directly address or relate to the user's query.

Rules:
- Extract entities in their smallest possible span
- Exclude all person names
- Focus only on entities crucial for answering the query
- Return entities exactly as they appear in the text

Background description:
{desc}
Query:
{query}

You will be given 3 examples to help you understand the task.
Example 1:
Background: "Hello, I'm Sarah. I work at Microsoft as a junior developer with 2 years of experience. I live in Seattle."
Query: "What skills should I focus on developing in my early tech career at a leading software company to advance from my entry-level programming role?"
Answer: ["Microsoft", "junior developer"]

[Additional examples omitted for brevity]

Your output will contain the following format:
Thought: Explain your reasoning step by step for selecting the relevant PII entities.
Answer: List the relevant PII entities, each enclosed in double quotes (""). Return only the list without explanation. Example: ["key_pii_1", ..., "key_pii_n"]

Please have your output follow the format below: (if there is only one entity, please output ["key_pii_1"]):
Thought: xxx
Answer: ["key_pii_1", ..., "key_pii_n"]

Figure 15: Prompt of Auto-CoT Method

Self-Consistency Query-related PII Detection Prompt

Please identify highly relevant PII (Personally Identifiable Information) entities from the background description PII entities that directly address or relate to the user's query.

Rules:
- Extract entities in their smallest possible span
- Exclude all person names
- Focus only on entities crucial for answering the query
- Return entities exactly as they appear in the text

Background description:
{desc}
Query:
{query}

Your output will contain the following format:
Thought: Generate 5 completely different perspectives of your reflections for selecting the relevant PII entities.
Summary: Output a summary of all your thinking.
Answer: List the relevant PII entities, each enclosed in double quotes (""). Return only the list without explanation. Example: ["key_pii_1", ..., "key_pii_n"]

Please have your output follow the format below: (if there is only one entity, please output ["key_pii_1"]):
Thought:
1. xxxxxx
2. xxxxxx
3. xxxxxx
4. xxxxxx
5. xxxxxx

Summary:
xxxxx

Answer: ["key_pii_1", ..., "key_pii_n"]

Figure 16: Prompt of Self-Consistency Method

Plan-and-Solve Query-related PII Detection Prompt

Please identify highly relevant PII (Personally Identifiable Information) entities from the background description that directly address or relate to the user's query.

Rules:
- Extract entities in their smallest possible span
- Exclude all person names
- Focus only on entities crucial for answering the query
- Return entities exactly as they appear in the text

Background description:
{desc}
Query:
{query}

Your output will contain the following format:
Thought: Please start with a general plan for selecting the relevant PII entities, and then think step-by-step how to solve it based on the plan.
Answer: List the relevant PII entities, each enclosed in double quotes (""). Return only the list without explanation. Example: ["key_pii_1", ..., "key_pii_n"]

Please have your output follow the format below: (if there is only one entity, please output ["key_pii_1"]):
Thought: xxx
Answer: ["key_pii_1", ..., "key_pii_n"]

Figure 17: Prompt of Plan and Solve CoT Method

LLM Judge Evaluation Prompt

I want you to act as a neutral judge evaluating responses to a user query. Your task is to determine which responses better address the user's intent.

User Query: {query}

Reference Response (from prompt with all information):
{original_response}

Response A (from prompt with all PII masked):
{masked_response}

Response B (from prompt with query-unrelated PII masked):
{adaptive_response}

Please evaluate which responses better satisfy the user's intent and need, ignoring the presence of personally identifiable information (PII). Focus only on how well each response answers the query.

Rate each response on a scale of 1-10 where 10 is perfect:
1. Response A rating (1-10):
2. Response B rating (1-10):

Then provide a final judgment comparing each masked response to the reference response with one of these options:
For Response A:
- Reference is much better than Response A
- Reference is slightly better than Response A
- Reference and Response A are equally good
- Response A is slightly better than Reference
- Response A is much better than Reference

For Response B:
- Reference is much better than Response B
- Reference is slightly better than Response B
- Reference and Response B are equally good
- Response B is slightly better than Reference
- Response B is much better than Reference

Provide your final judgments as:
JUDGMENT A: [your choice]
JUDGMENT B: [your choice]

Figure 18: Prompt of LLM-as-Judge