PII-TRACE: A Benchmark for Context-Aware PII Detection in Multi-Turn LLM Conversations

Kaiyuan Zhang[1,2*], Chuan Wang[1*], Joey Zhong[1], Paul Fryzel[1], Kyle Polley[1], Jerry Ma[1], and Ninghui Li[1,3],
[1]Perplexity, [2]Rutgers University, [3]Purdue University

Abstract

LLM assistants and agentic systems log long multi-turn conversations. AI providers often scan these conversations for Personally Identifiable Information (PII) and mask the PII before storing or processing conversation data. Yet most PII detectors and benchmarks target self-contained records rather than cross-turn evaluation. To evaluate PII detection across turns in multi-turn conversations, we introduce PII-TRACE (**T**racing **R**ecurring PII **A**cross **C**onversational **E**xchanges), to our knowledge the first PII benchmark to assess whether detectors identify PII in conversational contexts and cover every mention of a recurring identifier across turns. PII-TRACE contains 13,148 synthetic multi-turn dialogues in 13 languages with character-level spans and identifier clusters. Across eleven baselines, including frontier LLMs, no detector achieves full entity-level coverage without substantial false positives on PII-free conversations, and single-pass reading loses a third of the gold characters on long dialogues. To close this gap, we introduce PII-Tracer, a compact 0.6B-parameter detector trained with conversation-level supervision. PII-Tracer attains the highest entity-level coverage of any system we evaluate and also performs strongly on standard single-record benchmarks. We will release both the benchmark and detector upon publication.

1 Introduction

Scanning text for Personally Identifiable Information (PII) is a standard component of data pipelines in modern LLM systems. Providers detect and remove PII when curating pretraining corpora (Soldaini et al., 2024; Laurençon et al., 2022; Li et al., 2023; Grattafiori et al., 2024) and before storing or reusing collected data. This process helps satisfy data protection regulations (European Union, 2016) and reduces the risk that models memorize and later reveal sensitive information (Carlini et al., 2021). Increasingly, PII detectors are built on transformer-based language models, including a recently released open-weight bidirectional token classifier (OpenAI, 2026c).

Figure 1: One identifier (highlighted) recurs across a conversation. PII-TRACE's *consistent-detection* metric counts it as covered only if every mention is found.

Existing PII detectors are typically designed for relatively short, self-contained records rather than the long, multi-turn conversations common in LLM assistants and agentic systems. Within a single conversation, users may paste emails or medical reports, switch between languages, and interact with assistants that call external tools or share context with other agents (Yao et al., 2023; Xi et al., 2023; Packer et al., 2023). Detectors designed to process isolated records may therefore struggle with the length, complexity, and contextual dependencies of real-world conversations.

[*]Equal contribution.

Conversational context determines what counts as PII and which mentions must be detected. The same text span can be PII in one conversation but not in another. For example, a name is PII when it refers to the user, but not when it refers to a public figure or a place named after that person. Moreover, the same identifier may recur throughout a conversation. In Figure 1, for example, the name *Maria Torres* appears across multiple turns. A detector must identify every occurrence of a recurring identifier, as missing even one leaves that occurrence unprotected. This requirement becomes especially important when conversations are passed to external tools or other agents, where private content may be further exposed (Wang et al., 2025; Yagoubi et al., 2026). PII detection in conversations is therefore inherently a **context-aware** task.

However, established PII benchmarks are primarily document- or record-oriented rather than conversational (ai4Privacy, 2024; Pilán et al., 2022; Stubbs et al., 2015). In our experiments, we identify three properties that these benchmarks do not adequately evaluate. The first is *cross-turn consistency*: every mention of a recurring identifier should be detected. The second is avoiding *long-context degradation*, where recall decreases as conversation length grows (Liu et al., 2024; OpenAI, 2026c). The third is the ability to handle *multilingual mixed-medium text*, where languages may switch within a dialogue and prose may be interleaved with code and structured records. As a result, strong performance on existing benchmarks does not necessarily translate into effective PII detection in real-world conversational settings. The closest concurrent efforts, such as REDACT (Vats et al., 2026) and RedactionBench (Brynjólfsson et al., 2026), cover more languages and longer documents, but do not provide explicit cross-turn mention chains needed to evaluate cross-turn consistency.

This paper asks: *Do PII detectors consistently cover recurring identifiers in multi-turn LLM conversations?*

To answer this question, we introduce PII-TRACE, to our knowledge the first PII benchmark with an explicit cross-turn detection task for multi-turn conversations. PII-TRACE comprises 13,148 synthetic dialogues derived from production assistant traffic, with character-level annotations spanning nine identifier types. Mentions of the same identifier are further linked into entities, allowing us to evaluate whether a detector follows an identifier across the entire conversation. We call this property *consistent detection*: an entity is considered covered only if every mention of that entity is detected. Conversations in PII-TRACE range from fewer than 1,000 to more than 100,000 characters, enabling us to measure robustness to long-context degradation. The benchmark also includes conversations that switch languages and interleave prose with code, tables, and structured records, spanning 13 languages in total. It therefore enables evaluation of detectors on multilingual mixed-medium text. Finally, PII-TRACE includes PII-free conversations for measuring false positives.

We find that existing detectors struggle to consistently cover recurring identifiers across conversations. Across eleven baselines, ranging from open-source detectors to frontier LLMs prompted for PII detection, no public detector achieves strong consistent detection without flagging most PII-free conversations. Frontier LLMs are substantially more precise, but still miss many PII mentions.

To close this gap, we introduce PII-Tracer, a compact 0.6B-parameter detector trained on multilingual conversations. PII-Tracer labels an entire 4,096-token window in a single pass. Longer conversations can be processed using overlapping sliding windows, which substantially improves recall on long conversations (§4). PII-Tracer achieves the highest consistent detection and character-level F1 among all systems we evaluate, while using only a small fraction of the parameters of frontier LLMs (§3.7, §4).

Contributions.

- We introduce PII-TRACE, a multi-turn PII benchmark with an explicit cross-turn detection task. It includes long, multilingual, and mixed-medium conversations, with mentions linked into entities for evaluating *consistent detection* across turns.
- We evaluate eleven PII detectors, ranging from open-source specialized models to frontier LLMs. Public specialized detectors achieve high recall but low precision, while frontier LLMs are more precise but miss many mentions. For most detectors, recall also degrades substantially as conversations grow longer.

- We introduce PII-Tracer, a compact 0.6B-parameter PII detector trained on conversational data. It achieves better coverage of recurring identifiers than all other evaluated detectors, including much larger frontier LLMs.

2 Related Work

PII benchmarks. Existing benchmarks cover synthetic records, including ai4privacy, Nemotron-PII, and SPY (ai4Privacy, 2024; NVIDIA, 2025b; Savkin et al., 2025); legal documents with within-document coreference, such as TAB (Pilán et al., 2022); longitudinal but non-conversational clinical records, such as i2b2/UTHealth (Stubbs et al., 2015); and newswire named-entity recognition (NER), such as CoNLL-2003 (Tjong Kim Sang and De Meulder, 2003). Concurrent work includes REDACT, which covers mixed formats and multi-turn chats without cross-turn mention chains (Vats et al., 2026); the document-oriented RedactionBench (Brynjólfsson et al., 2026); and the query-focused PII-Bench and CAPID (Shen et al., 2026; Ponomarenko et al., 2026). None provides explicit mention chains across turns of a longitudinal user–assistant conversation (Table 1).

PII detectors. Existing detectors range from rule- and NER-based systems, such as Microsoft Presidio (Microsoft, 2024), to learned span models, including the OpenAI Privacy Filter (OpenAI, 2026c), GLiNER2-PII (Zaratiana et al., 2026), and open-source detectors from the Piiranha (iiiorg, 2024) and OpenMed (OpenMed Science, 2026) series. We evaluate seven of these detectors on conversations in §4.

Contextual and agentic privacy. Privacy is contextual: whether an information flow is appropriate depends on the context in which it occurs (Nissenbaum, 2004). LLMs can leak private information in inappropriate contexts (Mireshghallah et al., 2024), infer hidden attributes from contextual cues (Staab et al., 2024), and reproduce memorized training data (Carlini et al., 2021; Kim et al., 2023). Agentic privacy benchmarks test whether agent actions respect privacy norms (Shao et al., 2024) and measure information leakage through tool use and inter-agent interactions (Wang et al., 2025; Yagoubi et al., 2026). Separately, long-context studies show that model performance degrades as input length increases (Liu et al., 2024; Hsieh et al., 2024; Bai et al., 2024). PII-TRACE is complementary to this work: it evaluates span-level PII detection across an entire conversation.

3 The PII-TRACE Benchmark

3.1 Problem formulation

We study PII detection in long, multi-turn conversations. Given a complete user–assistant dialogue, the task is to identify every text span that refers to a private individual. We adopt an operational definition of PII as information that identifies a specific person, either on its own or when combined with other information. This definition draws on the GDPR (European Union, 2016), the ISO/IEC 29100 privacy framework (ISO/IEC, 2011), U.S. federal guidance (McCallister et al., 2010), and scholarship on contextual identifiability (Schwartz and Solove, 2011).

Consider the name *Maria Torres* in Figure 1. Whether this span constitutes PII cannot be determined from the string alone. If a user states their name, the span reveals personal information. The same string does not identify anyone when the assistant introduces it as a fictional placeholder in an example. Distinguishing among these cases requires reasoning about the surrounding conversational context, making PII detection a **context-aware** task.

Formally, a conversation is a sequence of m user and assistant turns concatenated into one text,

$$x \;=\; \tau_1 \,\|\, \tau_2 \,\|\, \cdots \,\|\, \tau_m \tag{1}$$

where turn τ_i occupies the character positions T_i of x. A detector is graded on all of x. It may read the text in one window or several (a choice we leave to its inference policy), but it must commit to a single set P of character positions marked as PII. We do not require it to say which marked positions belong to the same entity. The task is detection only.

The nine identifier types shown in Table 2 cover the label sets of deployed detectors (OpenAI, 2026c; Zaratiana et al., 2026). Let G be the set of gold PII characters. An *entity* $E \subseteq G$ is the set of characters belonging to a single identifier, with repeated mentions grouped by coreference, and the entities $\mathcal{E}$ partition G. We call an entity *cross-turn* if its characters appear in more than one turn in the conversation,

$$\left|\{\, i : E \cap T_i \neq \emptyset \,\}\right| \geq 2. \tag{2}$$

Table 1: PII-TRACE vs. representative PII benchmarks: only PII-TRACE pairs multi-turn conversational text with cross-turn identifier clusters (TAB annotates coreference within single documents; REDACT includes flat chats among mixed formats but defines no cross-turn task). Sizes approximate.

Benchmark	#Ex.	Langs	Setting	Entity-cluster	Cross-turn
ai4privacy (ai4Privacy, 2024)	225k	6	short document	✗	✗
Nemotron-PII (NVIDIA, 2025b)	100k	1	document	✗	✗
SPY (Savkin et al., 2025)	8.7k	1	QA (author)	✗	✗
TAB (Pilán et al., 2022)	1.3k	1	legal doc	✓	✗
REDACT (Vats et al., 2026)	13.4k	25	mixed incl. chat	✗	✗
PII-Bench (Shen et al., 2026)	2.8k	1	query + description	✗	✗
RedactionBench (Brynjólfsson et al., 2026)	0.2k	1	document	✗	✗
PII-TRACE (ours)	**13,148**	**13**	**multi-turn conv.**	✓	✓

Table 2: The nine identifier types, with gold mention counts over all three splits (37,431 total; every occurrence counts separately).

Type	Covers	Mentions
`private_person`	name of a private person: full name, name part, handle	22,973
`private_date`	date identifying a person	3,381
`private_url`	personal URL or IP address	2,657
`private_address`	postal address, precise location	2,521
`account_number`	national ID, SSN, IBAN, card, account ids	2,013
`private_email`	personal email address	1,626
`private_phone`	personal phone or fax	985
`other_pii`	residual annotator-marked identifiers	860
`secret`	credentials: passwords, keys, PINs	415

Achieving protection requires **consistent detection** because for recurring identifiers, missing even one occurrence leaves the identifier exposed. First, at the character level, a label is considered correct if a labeled position belongs to G. This measures how many PII texts are detected. Second, at the entity level, an entity is considered consistently detected if all characters of entity E are labeled ($E \subseteq P$). The evaluation setting (§4.1) embodies these two criteria into the precise metrics reported in this paper.

3.2 Design desiderata

Our design goals follow the actual deployment scenario: for the LLM assistant, the PII detector deals with dialogue logs, not the short, well-structured records common in existing corpora.

Splitting a dialogue into individual records results in the loss of significant structural information, since an identifier may reappear after many rounds. We assign a cluster ID to each occurrence, enabling evaluation to track the identifier across the entire dialogue rather than a single location. Dialogue lengths vary widely, from a few hundred to over one hundred thousand characters, allowing us to measure recall based on input length. Dialogues are also not always monolingual natural language text: they may switch languages mid-conversation, and plain text may contain code, tables, or structured records. In our experiments, existing baseline detectors performed poorly primarily on dialogues with these characteristics (§4).

Pipeline. We synthesize the benchmark from real traffic in five steps (Figure 2), labeling and anonymization (steps 1, 2, §3.3), synthesis (steps 3, 4, §3.4), and the verification gates that decide whether a record (step 5) is released (§3.5).

3.3 Labeling and anonymization

We begin with large-scale real-world user-assistant dialogue samples collected from a production environment, transforming each dialogue into a de-identified template. Multiple state-of-the-art LLMs are used to annotate identifiers according to the nine types of annotation in Table 2 (Figure 2, step 1). Subsequently, recurring identifiers of the same type are grouped into the same entity through rule-based processing, and sensitive attribute layers are annotated in a separate process. In step 2, we remove each identifier and replace it with a placeholder. The template retains the multi-turn structure of the dialogue, as well as the position, type, and entity ID of each occurrence, while not retaining any of the original annotated values.

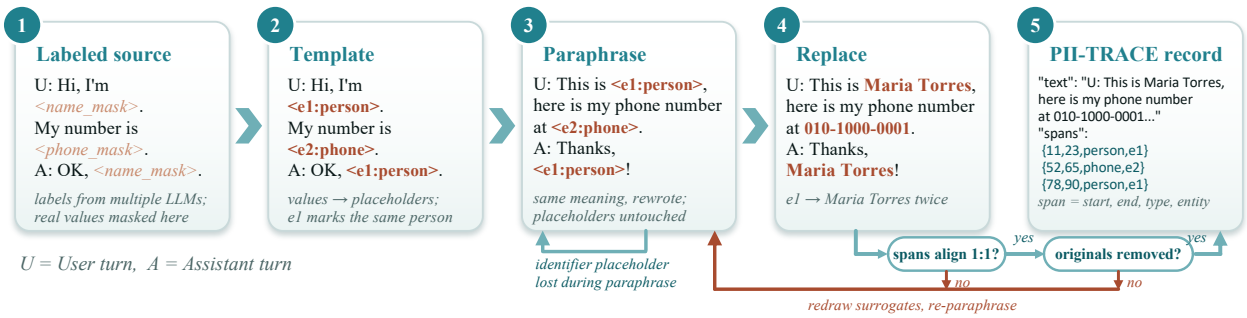

Figure 2: Constructing PII-TRACE, on one example (source values withheld): a labeled conversation becomes a typed, cluster-preserving template; each turn is paraphrased with placeholders intact, and each placeholder is replaced by its entity's surrogate with exact offsets. Failed paraphrases are retried, and gate-flagged documents can re-enter synthesis with fresh surrogates before a final drop (§3.5). Steps 1 and 2 are the anonymization of §3.3, steps 3 and 4 the synthesis of §3.4, and the gates, guarding the released record (step 5), the verification of §3.5.

3.4 Synthesis

Each template in the anonymization phase generates a synthesized dialogue (Figure 2, steps 3 and 4). Surrogates are generated for each entity, matching their type, format, and geographic location, and using values that are formatted correctly but not usable. The same surrogate is used for every occurrence, ensuring identifier consistency within a document, while surrogates in different documents are generated independently. The language model then rewrites each round of dialogue (step 3) and inserts the surrogates (step 4), so both the textual representation and the annotated identifier values in the final benchmark are synthesized content. If a placeholder is lost during paraphrasing, the rewrite is regenerated up to three times (back loop in step 3); if it still fails, the document is flagged and proceeds to the verification checks in §3.5. This replacement strategy follows the clinical de-identification practice (Carrell et al., 2013; Stubbs et al., 2015).

3.5 Alignment, verification, and audit

Every tagged identifier in the benchmark is a surrogate that the pipeline placed (Figure 2, step 5), so the gold span can be accurately determined without manual annotation. The alignment step further ensures accurate positioning: after rewriting, we recalculate each surrogate's character offset and append its type and entity ID.

Subsequently, the validation process verifies the correct construction of each document through three automated checks. First, the replacement value span must correspond one-to-one with the gold mention, and all mentions of the same entity must share the same value. Second, no original values should be detected during independent rescanning using Microsoft Presidio and strict regular expressions. Third, each stored character offset must accurately extract the corresponding surrogate substring. The first two checks correspond to the decision nodes in Figure 2. A document that fails any gate is resynthesized with fresh surrogates and re-paraphrased turns (the loop back into step 3), and discarded if it still fails. Since these checks are rule-based, we also used a second, independent language model to audit a portion of the released documents as a fuzzy check; anything it flagged was manually reviewed against the document's surrogate list. See Appendix E for the prompts used in the pipeline.

3.6 Composition

PII-TRACE contains 13,148 conversations, split into training, validation, and test sets with no source conversations shared between the different data partitions; 5,645 dialogues contain gold PIIs, and 7,503 are PII-free. Dialogue lengths range from less than 1,000 characters to over 100,000 characters, and thirteen languages each contain at least 100 dialogues. In dialogues containing PIIs, 63.8% of the dialogues contain entities mentioned multiple times, and 28.7% of the dialogues have entities that appear repeatedly across rounds; therefore, detecting only one mention often fails to cover the entire entity.

3.7 PII-Tracer: a context-aware detector

PII-Tracer models PII detection in the dialogue as token classification. One encoder pass reads a window (§3.1) of the dialogue x and generates contextual states $h_{1:T}$ for T tokens within it. Therefore, the annotation for each token is based on the surrounding dialogue window as context, rather than using the autoregressive prompting used in existing LLM baselines.

Architecture. PII-Tracer is a bidirectional encoder with 0.6B parameters, using a Qwen3 (Qwen Team, 2025) backbone adapted with masked-diffusion pretraining, reading a maximum of 4096 tokens per window. On the shared token states, the model uses a tagging head covering 37 BIOES labels (including a background tag O and {B, I, E, S} for each of the nine types), and an auxiliary sensitivity head used during training to predicts whether the conversation contains sensitive content as an auxiliary training signal. Gold spans are mapped to tokenization and represented as BIOES tags; therefore, detection is modeled as per-token classification with the surrounding conversation window as context.

Training and decoding. We minimize

$$\mathcal{L} = \lambda_{\text{tok}}\,\mathcal{L}_{\text{tag}} + \lambda_{\text{sens}}\,\mathcal{L}_{\text{sens}}, \tag{3}$$

where $\mathcal{L}_{\text{tag}}$ is the class-weighted cross-entropy for BIOES tags, and $\mathcal{L}_{\text{sens}}$ is the binary cross-entropy for sensitivity logit, with weights of λ_{tok}=1.5 and λ_{sens}=0.3, respectively. We trained on AdamW for 3 epochs with 714k samples, including multilingual assistant conversations annotated by multiple frontier language models and samples from the ai4privacy corpus. Each training sample concatenates all turns in a conversation into plain text and annotates each mention on the conversation-level offset, enabling the model to judge each span in conjunction with the conversational context. Therefore, the same string can be supervised as a PII in one conversation and as background in another. During inference, per-token scores are decoded into typed spans while ensuring the tag sequences are valid; two learnable boundary biases adjust the tradeoff between precision and recall without retraining; for conversations longer than the window, decoding can be performed window-by-window (§4). More hyperparameter settings in Appendix C.

4 Experiments

4.1 Evaluation setup

Our evaluation covers twelve detectors: Microsoft Presidio (Microsoft, 2024), six learned span models (the OpenAI Privacy Filter with 1.5B parameters (OpenAI, 2026c), GLiNER2-PII (Zaratiana et al., 2026), zero-shot GLiNER-PII (NVIDIA, 2025a), Piiranha (iiiorg, 2024), and two OpenMed clinical PII models (OpenMed Science, 2026)), four frontier LLMs prompted zero-shot with the nine-type taxonomy (GPT-5.4 (OpenAI, 2026a), GPT-5.6-sol (OpenAI, 2026b), Claude Opus 4.8 (Anthropic, 2026a), and Claude Sonnet 5 (Anthropic, 2026b)), and ours PII-Tracer. We evaluate on the test split (1,922 documents) and report metrics as following:

- **Character P / R / F1**: A predicted character is considered a true positive if it lies within a gold span. Character F1 is our primary detection metric, measuring how many PII texts are successfully masked without considering span boundaries.
- **Span-Overlap and Span-Containment P / R / F1**: For overlap, a prediction is considered correct if it overlaps with any gold span. For containment, precision counts predicted spans contained within a gold span, while recall counts gold spans contained within a predicted span.
- **Consistent detection**: The percentage of gold entities whose mentions are covered. CD-multi is calculated only for multi-mention entities, while cross-turn CD is calculated only for entities that repeat across rounds.
- **Has-PII accuracy and FP$_0$**: The former is the document-by-document binary has-PII accuracy, and the latter is the false-positive rate on documents without PII.

4.2 Main results

Table 3 divides the twelve systems into two groups. Publicly available specialized baselines achieve high recall through over-marking. OpenMed models can cover more than 0.9 of gold characters, but at the cost of marking more than six times the gold character volume, resulting in character precision below 0.15. Presidio marks 16 times the gold volume, with a precision of only 0.045, primarily covering structured

Table 3: PII detection on the PII-TRACE test set (1,922 documents); all metrics label-agnostic, higher is better. The four frontier LLMs are prompted zero-shot; span scores match gold spans by overlap and by containment.

	Character			Span-Overlap			Span-Containment		
Detector	P	R	F1	P	R	F1	P	R	F1
Presidio (rule/NER) (Microsoft, 2024)	0.045	0.762	0.086	0.042	0.844	0.079	0.036	0.741	0.069
OpenAI Privacy Filter (1.5B) (OpenAI, 2026c)	0.358	0.785	0.492	0.266	0.817	0.401	0.380	0.363	0.371
GLiNER2-PII (Zaratiana et al., 2026)	0.211	0.475	0.292	0.181	0.525	0.269	0.173	0.514	0.259
GLiNER-PII (zero-shot) (NVIDIA, 2025a)	0.261	0.785	0.392	0.209	0.890	0.338	0.272	0.483	0.348
Piiranha v1 (iiiorg, 2024)	0.161	0.299	0.209	0.117	0.339	0.174	0.078	0.157	0.104
OpenMed PII (44M) (OpenMed Science, 2026)	0.136	0.901	0.236	0.094	0.942	0.170	0.040	0.786	0.077
OpenMed PII (434M) (OpenMed Science, 2026)	0.145	0.919	0.251	0.109	0.957	0.195	0.047	0.787	0.089
GPT-5.4 (OpenAI, 2026a)	0.466	0.569	0.512	0.482	0.547	0.512	0.434	0.547	0.484
GPT-5.6-sol (OpenAI, 2026b)	0.555	0.679	0.611	0.589	0.681	0.632	0.558	0.679	0.612
Claude Opus 4.8 (Anthropic, 2026a)	0.502	0.556	0.528	0.488	0.519	0.503	0.470	0.515	0.491
Claude Sonnet 5 (Anthropic, 2026b)	0.536	0.602	0.567	0.579	0.566	0.572	0.510	0.570	0.539
PII-Tracer (0.6B)	0.507	0.830	0.629	0.496	0.832	0.621	0.445	0.831	0.580

Table 4: Consistent detection by an entity's mention count and, for multi-mention entities, by turn span: each cell is the fraction of that group's entities with *every* mention covered. Best per row in **bold**.

		Presidio	OpenAI PF	GLiNER2-PII	GPT		Claude		PII-Tracer
					5.4	5.6-sol	Opus 4.8	Sonnet 5	
count	1	0.655	0.474	0.641	0.829	0.788	0.840	0.829	**0.917**
	2	0.671	0.397	0.356	0.413	0.689	0.333	0.449	**0.873**
	3–5	0.630	0.263	0.258	0.216	0.504	0.216	0.252	**0.796**
	6–10	0.627	0.182	0.073	0.082	0.464	0.045	0.118	**0.691**
turn	cross-turn	0.648	0.296	0.292	0.275	0.551	0.222	0.305	**0.776**
	within-turn	0.615	0.343	0.154	0.302	0.663	0.320	0.355	**0.876**

identifiers such as email and phone numbers, with less coverage of free text. Frontier general-purpose LLMs exhibit the opposite pattern. They have the highest character precision, approaching 0.5, while all publicly available specialized baselines do not exceed 0.36; however, they only cover gold characters from 0.56 to 0.68.

PII-Tracer combines the advantages of both. It can cover 0.830 gold characters, comparable to over-marking detectors, while achieving a precision of 0.507, comparable to frontier LLMs; its char-F1 is 0.629, the highest of all systems, making it the most balanced among the twelve detectors. The much larger parameter-scale GPT-5.6-sol only slightly surpasses it in span-level F1 (overlap 0.632 vs. 0.621, containment 0.612 vs. 0.580).

4.3 Cross-turn consistency

Covering characters is easier than covering every mention of an entity, since missing any mention leaves the entity exposed; multi-mention consistent detection is therefore lower than character recall for all detectors. PII-Tracer covers 0.830 of gold characters, with a multi-mention consistent detection (CD-multi) of 0.794, the highest of all systems, significantly higher than GPT-5.6-sol (0.570) and other frontier LLMs (0.24-0.31). Aggregate CD, CD-multi, cross-turn CD, and PII-free false-positive rates for the twelve detectors are shown in Appendix D (Table 7), while Table 4 analyzes consistent detection by mention count (899 single-mention and 959 multi-mention entities, 790 of the latter appearing repeatedly across rounds). As the number of mentions increases, consistent detection declines sharply for learned detectors (GLiNER2-PII from 0.641 for a single mention to 0.073 for 6–10 mentions) and frontier LLMs (GPT-5.6-sol from 0.788 to 0.464, Claude Opus 4.8 from 0.840 to 0.045), whereas PII-Tracer declines more gradually (from 0.917 to 0.691) and performs best across all groups, including cross-turn entities (0.776). Presidio consistently performs at

Table 5: PII-Tracer character P/R/F1 by conversation length, read in a single 4,096-token window. Docs: test documents per bucket; PII docs: those with gold spans; characters pooled per bucket.

Length	Docs	PII docs	P	R	F1
<1k	167	49	0.392	0.975	0.559
1k–10k	1,292	505	0.516	0.955	0.670
≥10k	463	243	0.507	0.687	0.583
Total	1,922	797	0.507	0.830	0.629

around 0.63 across all groups because it labels most structured values rather than selectively covering a particular entity.

4.4 Document-level flagging

In practice, Guardrail must also determine whether a conversation contains PII, as incorrectly labeling clean conversations incurs costs. Has-PII accuracy reveals over-flagging issues: Presidio and OpenMed models score between 0.43 and 0.45, below the Has-PII baseline because they label PII in almost every conversation; frontier LLMs score between 0.76 and 0.83; and PII-Tracer reaches 0.759, the highest among specialized detectors and close to frontier LLMs. The complementary metric FP_0 is shown in Table 7: all public specialized detectors label more than half of the PII-free documents, while the proportion for PII-Tracer is 0.385. This reflects the cost of over-marking at the document level, as predicted spans scattered throughout clean conversations become false flags.

4.5 Single-window degradation

As dialogue length increases, recall decreases. In a single 4096-token window, PII-Tracer covers 0.975 of gold characters in dialogues shorter than 1k characters, 0.955 in dialogues between 1 and 10k characters, and drops to 0.687 for dialogues longer than 10k characters (Table 5). Precision does not decrease with length; it is lowest in the shortest range (0.392), and around 0.51 in other ranges. This is primarily due to single-window truncation: using 50%-overlap sliding windows decoding for the same checkpoint can improve character recall from 0.83 to 0.97 and multi-mention CD from 0.79 to 0.95 (Appendix D). Longer dialogues contain a larger absolute number of identifiers, so documents with decreased recall face a greater risk of missed detections. The sliding-window method incurs a slight loss of precision, but the two strategies can be chosen at decode time, so a suitable operating point can be selected during deployment without retraining.

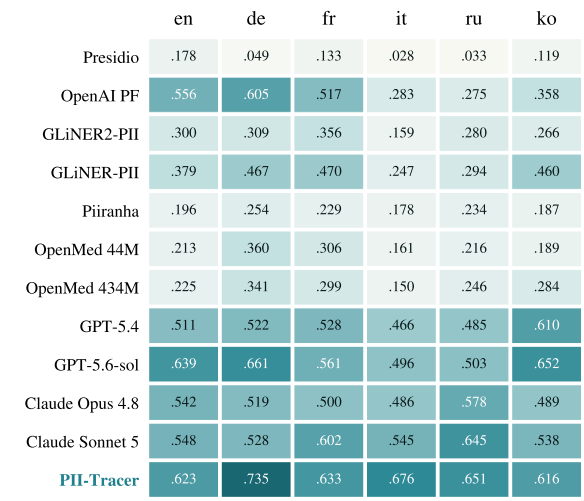

Figure 3: Character F1 by language for six test languages covering the Latin, Cyrillic, and Hangul scripts.

4.6 Multilingual mixed-medium text

Figure 3 shows the character F1 scores for six language subsets in the corpus, covering Latin, Cyrillic, and Hangul scripts; full language-specific results will be available upon release. Each language was evaluated on all test documents, ranging from 928 for English to 61 for Italian. System rankings varied across different scripts. Presidio covered 0.79 of gold characters on English but only 0.42 on Russian; OpenMed 434M's recall dropped from 0.94 for English to 0.69 for Korean; OpenAI Privacy Filter maintained recall across different scripts, thus its F1 score dropped to 0.28 on Italian and Russian, primarily due to precision. PII-Tracer achieved the highest character F1 score in four of the six languages (all between 0.616 and 0.735; GPT-5.6-sol scored slightly higher in 'en' and 'ko'), and the highest consistent detection across all six languages, ranging from 0.80 to 0.93. Because rankings vary across different scripts, selecting a detector solely based on overall or English scores may not suit for real-world language combinations.

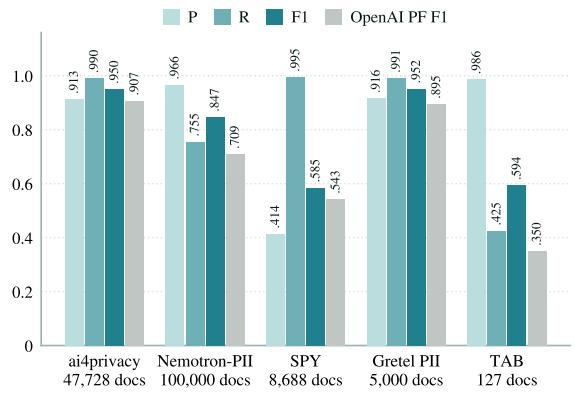

Figure 4: PII-Tracer on five external PII benchmarks (label-agnostic character P/R/F1); gray: the OpenAI Privacy Filter under identical evaluation.

4.7 External benchmarks

Figure 4 shows the results of PII-Tracer on five external PII benchmarks, including two public PII datasets used in the OpenAI Privacy Filter model card: ai4privacy and SPY. PII-Tracer outperforms the Privacy Filter on all benchmarks: char F1 0.950 vs 0.907 on ai4privacy (ai4Privacy, 2024), 0.847 vs 0.709 on Nemotron-PII (NVIDIA, 2025b), 0.585 vs 0.543 on SPY (Savkin et al., 2025), 0.952 vs 0.895 on the Gretel PII masking set (AI, 2024), and 0.594 vs 0.350 on TAB (Pilán et al., 2022). TAB is the only benchmark that includes real human-annotated text, and at the same precision, PII-Tracer achieves twice the recall. The improvement does not stem from a shared synthesis style. Table 3 indicates that single-record detectors perform poorly in dialogue scenarios, whereas PII-Tracer is competitive in both settings. Since its training mixture includes single-record data, we do not attribute this result entirely to conversational supervision.

5 Conclusion

In this work, we propose PII-TRACE, a benchmark for PII detection in multi-turn LLM conversations, and PII-Tracer, a compact 0.6B detector trained on such data. Experiments show that existing detectors either over-label PII-free conversations or miss some mentions of recurring identifiers; in contrast, PII-Tracer more completely covers recurring identifiers and performs well on standard external benchmarks. We call for greater attention to PII detection in multi-turn LLM conversations.

Limitations

PII-TRACE is designed as a controlled benchmark of PII detection in multi-turn user–assistant conversations. Its conversations are synthetic reconstructions derived from the structure of production assistant traffic. This construction supports public release with exact span offsets and consistent mention chains, but it does not reproduce the source distribution verbatim. The reported results should therefore be read as measurements of conversational PII detection rather than estimates for any particular production workload.

The current release focuses on conversation text. Tool calls, inter-agent messages, and multimodal inputs fall outside its scope, although they are natural extensions for studying how personal data moves through agentic systems. Evaluation on conversations from additional assistants and domains would also provide a broader test of transfer beyond the setting represented here.

Finally, each baseline is evaluated under a single inference configuration. Alternative prompts, thresholds, context-window policies, or future model updates may change absolute scores. The benchmark is thus best suited to comparing detector behavior within the evaluation setup used here, rather than certifying production safety or regulatory compliance.

Ethical Considerations

PII-TRACE is built to strengthen protective systems: the intended use is evaluating and improving PII detectors, not re-identifying anyone. The source conversations were accessed and processed under the originating provider's terms of service and privacy policy; access was limited to authorized project members under the provider's data-access controls, and the manual review of audit-flagged documents took place under the same controls. No crowdworkers or external annotators were involved: identifiers were labeled by prompted language models, and human review was limited to the project team. We release only synthetic data: no source conversation is published, every tagged identifier is a fabricated, leak-checked surrogate, and the known residual, untagged personal names in some non-English text, is documented in the datasheet (Appendix A). The benchmark and the de-

tector will be released under the MIT license. Scores on synthetic data are proxies for, not guarantees of, behavior on real traffic, and the benchmark certifies neither production safety nor regulatory compliance. Language models assisted with writing and engineering in this project; the authors reviewed and verified all content and results.

References

Gretel AI. 2024. Gliner models for pii detection through fine-tuning on gretel-generated synthetic documents.

ai4Privacy. 2024. PII-masking-300k. https://huggingface.co/datasets/ai4privacy/pii-masking-300k.

Anthropic. 2026a. System card: Claude Opus 4.8. https://www.anthropic.com/news/claude-opus-4-8. May 28, 2026.

Anthropic. 2026b. System card: Claude Sonnet 5. https://www.anthropic.com/news/claude-sonnet-5. June 30, 2026.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024. LongBench: A bilingual, multitask benchmark for long context understanding. In *Annual Meeting of the Association for Computational Linguistics (ACL)*. ArXiv:2308.14508.

Sean Brynjólfsson, Shashvat Jayakrishnan, Esha Sali, Diptanshu Purwar, and Madhav Aggarwal. 2026. RedactionBench. *arXiv preprint arXiv:2606.18782*. ArXiv:2606.18782.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. 2021. Extracting training data from large language models. In *USENIX Security Symposium*. ArXiv:2012.07805.

David Carrell, Bradley Malin, John Aberdeen, Samuel Bayer, Cheryl Clark, Ben Wellner, and Lynette Hirschman. 2013. Hiding in plain sight: use of realistic surrogates to reduce exposure of protected health information in clinical text. *Journal of the American Medical Informatics Association*, 20(2).

European Union. 2016. Regulation (eu) 2016/679 of the european parliament and of the council (general data protection regulation). Official Journal of the European Union, L 119.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. 2021. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92. ArXiv:1803.09010.

Aaron Grattafiori et al. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. RULER: What's the real context size of your long-context language models? In *Conference on Language Modeling (COLM)*. ArXiv:2404.06654.

iiiorg. 2024. Piiranha-v1: Detect personal information. https://huggingface.co/iiiorg/piiranha-v1-detect-personal-information. Accessed 2026.

ISO/IEC. 2011. Iso/iec 29100:2011 information technology – security techniques – privacy framework. International Organization for Standardization.

Siwon Kim, Sangdoo Yun, Hwaran Lee, Martin Gubri, Sungroh Yoon, and Seong Joon Oh. 2023. ProPILE: Probing privacy leakage in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*. ArXiv:2307.01881.

Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, et al. 2022. The BigScience ROOTS corpus: A 1.6TB composite multilingual dataset. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*. ArXiv:2303.03915.

Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, et al. 2023. StarCoder: May the source be with you! *Transactions on Machine Learning Research*. ArXiv:2305.06161.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics (TACL)*, 12:157–173. ArXiv:2307.03172.

Erika McCallister, Tim Grance, and Karen Scarfone. 2010. Guide to protecting the confidentiality of personally identifiable information (pii). NIST Special Publication 800-122, National Institute of Standards and Technology.

Microsoft. 2024. Presidio: Data protection and de-identification sdk. https://github.com/microsoft/presidio. Accessed 2026.

Niloofar Mireshghallah, Hyunwoo Kim, Xuhui Zhou, Yulia Tsvetkov, Maarten Sap, Reza Shokri, and Yejin Choi. 2024. Can LLMs keep a secret? testing privacy implications of language models via contextual integrity theory. In *International Conference on Learning Representations (ICLR)*. Spotlight. arXiv:2310.17884.

Helen Nissenbaum. 2004. Privacy as contextual integrity. *Washington Law Review*, 79(1):119–157.

NVIDIA. 2025a. GLiNER-PII: Zero-shot PII named entity recognition. https://huggingface.co/nvidia/gliner-PII. Accessed 2026.

NVIDIA. 2025b. Nemotron-PII. https://huggingface.co/datasets/nvidia/Nemotron-PII.

OpenAI. 2026a. GPT-5.4 Thinking system card. https://deploymentsafety.openai.com/gpt-5-4-thinking. March 5, 2026.

OpenAI. 2026b. GPT-5.6 Preview system card. https://deploymentsafety.openai.com/gpt-5-6-preview. June 25, 2026; covers the Sol, Terra, and Luna models.

OpenAI. 2026c. Model card for OpenAI privacy filter. https://cdn.openai.com/pdf/c66281ed-b638-456a-8ce1-97e9f5264a90/OpenAI-Privacy-Filter-Model-Card.pdf. Accessed 2026.

OpenMed Science. 2026. OpenMed-PII-SuperClinical: Pii detection models. https://huggingface.co/OpenMed/OpenMed-PII-SuperClinical-Small-44M-v1. Accessed 2026.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. 2023. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*. ArXiv:2310.08560.

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. ArXiv:2202.00443.

Mariia Ponomarenko, Sepideh Abedini, Masoumeh Shafieinejad, D. B. Emerson, Shubhankar Mohapatra, and Xi He. 2026. CAPID: Context-aware PII detection for question-answering systems. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 320–331. ArXiv:2602.10074.

Qwen Team. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Maksim Savkin, Timur Ionov, and Vasily Konovalov. 2025. SPY: Enhancing privacy with synthetic PII detection dataset. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*, pages 236–246. ACL Anthology 2025.naacl-srw.23.

Paul M. Schwartz and Daniel J. Solove. 2011. The pii problem: Privacy and a new concept of personally identifiable information. *New York University Law Review*, 86:1814–1894.

Yijia Shao, Tianshi Li, Weiyan Shi, Yanchen Liu, and Diyi Yang. 2024. PrivacyLens: Evaluating privacy norm awareness of language models in action. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*. ArXiv:2409.00138.

Hao Shen, Zhouhong Gu, Haokai Hong, Weili Han, and Hongfeng Chai. 2026. PII-Bench: Evaluating query-aware privacy protection systems. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4991–5026. ArXiv:2502.18545.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, et al. 2024. Dolma: An open corpus of three trillion tokens for language model pretraining research. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15725–15788. ArXiv:2402.00159.

Robin Staab, Mark Vero, Mislav Balunović, and Martin Vechev. 2024. Beyond memorization: Violating privacy via inference with large language models. In *International Conference on Learning Representations (ICLR)*. ArXiv:2310.07298.

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.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147. ArXiv:cs/0306050.

Guneesh Vats, Anubha Agrawal, Shikha Singhal, Ajita Dash, Praison Selvaraj, Vidhan Jhawar, Ranga Prasad Chenna, and Bharadwaj Y M G. 2026. Redact: A systematically controlled multilingual benchmark for personal information detection. *arXiv preprint arXiv:2606.19881*.

Shouju Wang, Fenglin Yu, Xirui Liu, Xiaoting Qin, Jue Zhang, Qingwei Lin, Dongmei Zhang, and Saravan Rajmohan. 2025. Privacy in action: Towards realistic privacy mitigation and evaluation for LLM-powered agents. In *Findings of the Association for Computational Linguistics: EMNLP*. ArXiv:2509.17488.

Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, Rui Zheng, Xiaoran Fan, Xiao Wang, Limao Xiong, Yuhao Zhou, Weiran Wang, Changhao Jiang, Yicheng Zou, Xiangyang Liu, and 10 others. 2023. The rise and potential of large language model based agents: A survey. *arXiv preprint arXiv:2309.07864*. ArXiv:2309.07864.

Faouzi El Yagoubi, Godwin Badu-Marfo, and Ranwa Al Mallah. 2026. AgentLeak: A benchmark for internal-channel privacy leakage in multi-agent LLM systems. ArXiv:2602.11510.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*. ArXiv:2210.03629.

Urchade Zaratiana, Ash Lewis, and George Hurn-Maloney. 2026. GLiNER2-PII: A multilingual model for personally identifiable information extraction. *arXiv preprint arXiv:2605.09973*. ArXiv:2605.09973.

A Datasheet for Datasets

Following Gebru et al. (2021), we answer the core questions inline; the release will include a data card with the full questionnaire.

Motivation. PII-TRACE was created to evaluate PII detection *in multi-turn conversational context*, an axis missing from prior sentence- or record-level resources.

Composition. Each instance is a synthetic multi-turn user–assistant conversation with character-level identifier mentions over nine types, per-mention cluster ids, and a separate sensitive-attribute layer. The corpus has 13,148 conversations across 13 languages (train/validation/test = 9,202 / 2,024 / 1,922), with per-record metadata. Figure 5 shows a complete released record.

Collection and annotation. The corpus is derived from a deduplicated sample of real user–assistant conversations from a production assistant; a "turn" is one delivered message. To anonymize each conversation, multiple frontier LLMs locate the identifiers, following a written guideline, and a rule-based pass groups repeated mentions of the same typed identifier into an entity; sensitive attributes are marked separately. We inherit these labels as PII-TRACE's gold. The labeler prompt and guideline are summarized in Appendix E, and the release will include them.

B Full Taxonomy and Crosswalk

Table 6 lists the nine identifier types with their subtypes and definitions, plus a crosswalk to standard regulatory categories; the full row-by-row mapping to OMB, NIST SP 800-122, ISO/IEC 29100, and HIPAA will accompany the release. The label space subsumes those of the deployed detectors we evaluate, so their outputs map directly into our types; the residual `other_pii` covers identifiers outside the eight concrete types. The eight sensitive-attribute classes (health condition, religion, race/ethnicity, sexuality, political view, disability, age, gender) form the separate attribute layer and are not part of the identifier taxonomy.

C Detector: Additional Details

Training details. We train with AdamW (learning rate 10^{-4}, effective batch 256) on 8 data-parallel GPUs for 3 epochs, selecting the best checkpoint on held-out loss. The training data and PII-TRACE's sources share the same production traffic and matching conversation ids.

Frontier-LLM detector protocol. The four frontier LLMs in Table 3 are prompted zero-shot, with no few-shot examples. A system message states the task and the nine identifier types and asks for a JSON list of {`text`, `type`} objects whose `text` is copied verbatim from the input (Appendix E). Documents are read in 8,000-character windows with 400-character overlap; each returned string is mapped back to character offsets by exact substring match, and offsets are merged and de-duplicated across windows.

D Extended Results

Full detection results. Table 7 completes the entity-level view of §4.3 for all twelve detectors. PII-Tracer leads on all three coverage columns (CD 0.853, CD-m 0.794, xCD 0.776). Its FP_0 of 0.385 is comparable to the frontier LLMs (0.231–0.386) and

```
{"id": "conv_00042", "language": "en", "length_bucket": "<1k",
 "document_format": "unstructured", "clean_negative": false,
 "text": "U: I'm Dana Okoye, my number is +00 10 1111 0000.\nA: Thanks, Dana Okoye, I'
ve noted the callback line.\nU: Please keep Dana Okoye on file.",
 "spans": [
   {"start": 7,   "end": 17,  "label": "private_person", "subtype": "full_name", "
entity_id": 1, "entity_mentions": 3},
   {"start": 32,  "end": 48,  "label": "private_phone",  "subtype": "phone",     "
entity_id": 2, "entity_mentions": 1},
   {"start": 61,  "end": 71,  "label": "private_person", "subtype": "full_name", "
entity_id": 1, "entity_mentions": 3},
   {"start": 118, "end": 128, "label": "private_person", "subtype": "full_name", "
entity_id": 1, "entity_mentions": 3}],
 "attribute_spans": []}
```


Figure 5: A complete released record (values fabricated). Offsets index the `text` field; `entity_id` links repeated mentions of one identifier, and `entity_mentions` is the cluster size used by consistent detection.

Table 6: The nine PII-TRACE identifier types: sub-types, one-line definition (a span is labeled only when it identifies a *private* person; placeholders and public entities are excluded), and regulatory crosswalk.

Type	Sub-types	Definition / crosswalk
`private_person`	full name, name part, social handle/username	name of a private person; OMB/NIST direct, HIPAA name
`private_email`	email address	personal email; NIST, HIPAA electronic mail
`private_phone`	phone/fax number	phone of a private person; NIST, HIPAA
`private_address`	postal address, precise geo	location of a private person; HIPAA geographic
`private_url`	profile URL, IP address	URL/IP for a private audience; NIST online identifier
`private_date`	date of birth, identifying date	datetime identifying a person; HIPAA dates
`account_number`	national ID, SSN, IBAN, credit card, sort code, id number	account/record identifier; NIST, HIPAA account no.
`secret`	password/API key, CVV/PIN	credential; NIST authentication data
`other_pii`	residual annotator-marked identifier	person-identifying but outside the eight types

well below every public specialized baseline (0.557–0.972). Coverage and false positives must be read together, since a detector can raise CD by flagging nearly every document, as Presidio and the OpenMed models do.

Inference policy for long documents. Table 8 decodes the same PII-Tracer checkpoint under three long-input policies. Relative to the single window, sliding windows raise recall on documents at or above 10k characters from 0.687 to 0.975 and CD-m from 0.794 to 0.954, while character precision drops from 0.507 to 0.493; chunked decoding lies between the two.

E Prompts

The four LLM prompts of this work are shown below.

Gold Labeler Prompt (§3.3)

Task. You are an expert PII labeler producing gold labels for … conversations. Your labels will train a downstream PII-masking model, so they must reflect industry-standard definitions of Personally Identifiable Information …

Test. Would a knowledgeable reader, given this string and the surrounding context, be able to identify or contact a specific person? If the string instead identifies a publicly-listed organization, a public-figure persona, or a fictional vignette persona, the answer is no …

Traps. Even when the string looks exactly like a name, address, phone, URL, or date, emit nothing if the context matches a trap: a named

Table 7: Entity-level results on the test split: consistent detection (CD), multi-mention CD (CD-m), cross-turn CD (xCD), and PII-free false-positive rate FP_0 (lower is better). Best per column in **bold**.

Detector	CD	CD-m	xCD	FP_0
Presidio	0.649	0.642	0.648	0.972
OpenAI PF	0.386	0.304	0.296	0.557
GLiNER2-PII	0.448	0.268	0.292	0.862
GLiNER-PII	0.472	0.349	0.334	0.901
Piiranha	0.128	0.090	0.097	0.685
OpenMed 44M	0.724	0.745	0.761	0.936
OpenMed 434M	0.709	0.743	0.759	0.935
GPT-5.4	0.545	0.279	0.275	0.321
GPT-5.6-sol	0.675	0.570	0.551	**0.231**
Claude Opus 4.8	0.530	0.239	0.222	0.386
Claude Sonnet 5	0.563	0.314	0.305	0.276
PII-Tracer	**0.853**	**0.794**	**0.776**	0.385

Table 8: PII-Tracer under three long-input decoding policies: a single window (trunc), non-overlapping windows (chunk), and 50%-overlap sliding windows (slide). $R_{<10k}$ and $R_{\geq 10k}$ split character recall at 10k characters.

Policy	P	R	F1	CD	CD-m	$R_{<10k}$	$R_{\geq 10k}$
trunc	0.507	0.830	0.629	0.853	0.794	0.956	0.687
chunk	0.502	0.944	0.656	0.916	0.894	0.956	0.931
slide	0.493	0.965	0.653	0.954	0.954	0.957	0.975

officer or staff member in an organizational role, reserved or documentation values, citation or tool tokens . . .

Output. A single JSON object with two fields: a conversation-level sensitivity head and the token-level PII span list of {`label`, `text`} objects . . .

Paraphrase Prompt (Figure 2, step 3)

Task. Rewrite a snippet from a real user–assistant chat so it is not verbatim the original, while preserving its meaning, intent, tone, language, and Markdown structure (headings, lists, tables, code fences . . .).

Rule 1: placeholders. The text may contain special placeholder characters; each stands for a redacted value. Reproduce every placeholder exactly once, in the natural grammatical position where its value belongs; never delete, duplicate, translate . . .

Rule 2: de-identify context. Any real-world specific that is not a placeholder (company, product, or brand names; unusual place names; distinctive verbatim phrases . . .) must become a generic or plausibly invented equivalent.

Output. Roughly the same length and the same language as the input; only the rewritten snippet, no preamble . . .

Second-LLM Audit Prompt (§3.5)

Task. Audit synthetic-conversation text for a PII benchmark; all names, contacts, and IDs are supposed to be fabricated . . .

Report. Only strings that look like a real, specific individual's genuine identifying data that fabrication might have missed (an intact real email, phone, or handle; a real public figure named as the actual user . . .).

Ignore. Generic names, places, and dates; clearly synthetic values.

Output. JSON `{"suspect": bool, "items": [{"text", "why"} ...]}`.

Frontier-LLM Detector Prompt (§4)

Task. You are a precise PII detector operating on text from a multi-turn assistant conversation. Extract every span that is personal information identifying a private individual . . .

Rule. Mark a span only when, in this conversational context, it helps identify a private person; do not mark public figures, organizations, fictional or placeholder values . . . A date counts only when it identifies a person . . .

Output. A JSON list of {`text`, `type`} objects over the nine identifier types, with `text` copied verbatim from the input . . .