USENIX Security '25 Artifact Appendix: Evaluating LLM-based Personal Information Extraction and Countermeasures

Yupei Liu, Yuqi Jia, Jinyuan Jia, and Neil Zhenqiang Gong

A Artifact Appendix

This is the artifact appendix for our USENIX Security'25 paper. This appendix is meant to describe a roadmap for the readers to understand what our artifact is and how to properly use our artifact, including any necessary environment.

A.1 Abstract

Our artifact contains three components: 1) the code for main experiments, 2) the code for baseline comparison, and 3) the Conda environment we used for evaluation.

A.2 Description & Requirements

In this section, we describe details of how to access our artifact, the hardware/software dependencies of our artifact, and any benchmarks in our artifact.

A.2.1 Security, privacy, and ethical concerns

As discussed in the Phase-1 AE, README in the "official_code" directory in the Zenodo URL, and the appendix of our paper, we will not share the other 3 datasets (i.e., Celebrity, Physician, and Professor) by default. Instead, we only share them per request to verified researchers. This is due to the dual nature of the release of these real-world datasets. In particular, they can benefit both defenders and attackers. Therefore, in this artifact, we don't provide the data or any configuration/code related to these 3 datasets.

A.2.2 How to access

The URL to access our main artifact is: https://zenodo.org/records/14737200. This repository contains three files/directories at root: 1) README, 2) official_code which is the code/data for our main experiments, and 3) baseline_code which contains our implementation to the baseline methods compared in our paper.

In addition, the URL to the Conda environment we used for our experiments is available at: https://zenodo.org/records/14868933.

A.2.3 Hardware dependencies

We conducted most of experiments on a single Quadro RTX 6000 GPU. Less powerful GPUs should also be able to run some experiments using our artifact on smaller LLMs. We note that for Flan-UL2 experiments, we use a single RTX Titan GPU. Regarding open-source models studied in our paper and released in our artifact, they can be executed even with only fairly poor computational resources like CPUs, because these models' results are obtained by querying remote models using their APIs. Moreover, we want to note that we used x86_64 platform for our experiments. To run the artifact on other archs like aarch64, the libraries may differ.

A.2.4 Software dependencies

For experiments on open-source LLMs, third-party softwares or APIs are required. For Google's Gemini pro, an account to Google's cloud platform is required. For OpenAI's GPT models, we use Microsoft Azure APIs for deployment and querying. For PaLM2 models, we leveraged Google's cloud APIs when we did the experiments. However, we note that PaLM2 models are no longer available on Google's platforms now. Instead, users may leverage a third-party deployment such as VertexAI or ClarifAI.

The URL to the Conda environment we used for our evaluation is at: https://zenodo.org/records/14868933. We use glnxa64 and our CUDA version is 12.1.

A.2.5 Benchmarks

Our paper does not produce any new models in our artifact. However, we indeed released the Synthetic dataset used in our evaluation. It is described in the README of the "official_code" directory. In addition, we provide the code to download the "Court" dataset which is from a previous work.

Again, we want to clarify that for ethical concerns, we don't release the other three datasets (i.e., Celebrity, Physician, and Professor) by default, because these datasets contain real-world persons' information. They are only shared per request with verified researchers. Future researchers who are interesed in obtaining access to these datasets are expected to contact the authors using institutional email addresses and explain the main purpose of using these datasets. This way of sharing these three real-world datasets, as discussed previously with

reviewers, is to guarantee that they are not misused by any potential malicious parties.

A.3 Set-up

The setup should involve preparing the right hardware such as GPUs for the artifact. Please refer to A.2.3 for more details.

A.3.1 Installation

Users can first download the "PIE_environment.yml" from the URL mentioned above. Then, users can use Conda command such as "conda env create –name envname –file=PIE_environment.yml" to create a virtual environment from this file. Next, users can download the code and data from https://zenodo.org/records/14737200 and extract the .zip files.

A.3.2 Basic Test

Users can step into "official_code" directory and run "python3 run.py" for a basic test. Note that this will run the Vicuna-13b-v1.3 model on the Synthetic dataset. If users want to try smaller LLMs, they can edit the "run.py" to disable Vicuna-13b-v1.3 and enable another LLM instead.

For the basic test, users may need to change the paths in the artifact.

A.4 Evaluation workflow

A.4.1 Major Claims

Our paper has the following major claims:

(C1): *LLMs can effectively extract personal information from personal profiles. This corresponds to E1.*
(C2): *Existing defenses against PIE are not as effective as prompt injection as a defense. This corresponds to E2.*

A.4.2 Experiments

(E1): *[LLM PIE Extraction] [30 human-minutes + 20 compute-hour + 3GB disk]: this experiment evaluates how accurate LLMs can extract PII from profiles.*
How to: *Use official_code/run.py to run this experiments. The run.py is self-explanatory and contains configurable parts where users can enable/disable. After running run.py, users can use run_evaluate.py to generate evaluation results.*
Preparation: *The virtual environment described in above sections should be enabled. Depending on hardware, more packages may be required to be installed in the actual requirements. Please follow the error messages to install any required packages.*
Execution: *First, cd into official_code. Then, run python3 run.py*
Results: *The results will be in official_code/logs.*
(E2): *[Defense evaluation] [30 human-minutes + 15 compute-hour]: this experiment evaluates the defense effectiveness against LLM-based PIE.*
How to: *Similarly, users can use official_code/run.py to run this experiments by changing the value of the variable "defenses" to a list consisting of names of defenses. The run.py is self-explanatory and contains configurable parts where users can enable/disable. After running run.py, users can use run_evaluate.py to generate evaluation results.*
Preparation: *The virtual environment described in above sections should be enabled. Depending on hardware, more packages may be required to be installed in the actual requirements. Please follow the error messages to install any required packages.*
Execution: *First, cd into official_code. Then, adjust run.py to change the "defenses". Next, run python3 run.py*
Results: *The results will be in official_code/logs.*

A.5 Version

Based on the LaTeX template for Artifact Evaluation V20231005. Submission, reviewing and badging methodology followed for the evaluation of this artifact can be found at https://secartifacts.github.io/usenixsec2025/.