The Journal of Open Source Software

PII-Codex: a Python library for PII detection, categorization, and severity assessment

Eidan J. Rosado [1]

1 College of Computing and Engineering, Nova Southeastern University, Fort Lauderdale, FL 33314, USA

DOI: 10.21105/joss.05402

Software

- Review
- Repository
- Archive

Editor: Arfon Smith

Reviewers:

- @gradvohl
- @tmickleydoyle

Submitted: 30 December 2022
Published: 20 June 2023

Summary

There have been a number of advancements in the detection of personal identifiable information (PII) and scrubbing libraries to aid developers and researchers in their detection and anonymization efforts. With the recent shift in data handling procedures and global policy implementations regarding identifying information, it is becoming more important for data consumers to be aware of what data needs to be scrubbed, why it's being scrubbed, and to have the means to perform said scrubbing.

PII-Codex is a collection of extended theoretical, conceptual, and policy works in PII categorization and severity assessment (Milne et al., 2016; Schwartz & Solove, 2011), and the integration thereof with PII detection software and API client adapters. It allows researchers to analyze a body of text or a collection thereof and determine whether the PII detected within these texts, if any, are considered identifiable. Furthermore, it allows end-users to determine the severity and associated categorizations of detected PII tokens.

Challenges

While a number of open-source PII detection libraries have been created and PII detection APIs are provided by cloud service providers (Azure, 2022; Services, 2022), the detection results are often provided with the type of PII detected, an index reference of where the detection is within the text, and a confidence score associated with the detection. Those receiving these results aren't provided with a means of understanding why the text token is classified as PII, what framework, policy, or convention labels it as such, and just how severe its exposure is.

Statement of Need

The general knowledge base of identifiable data, the usage restrictions of this data, and the associated policies surrounding it have shifted drastically over the years. Between the mid-1990s and 2000s, or the dotcom bubble, the industry saw a rise in data capitalism by way of making information freely accessible, fostering a way to make the web personal, and finally, placing value on data and the potential it had to impact consumerism (West, 2017). Alongside the rise in data capitalism came early data policy initiatives. In 1995, the EU Data Protection Directive was created to establish some minimum data privacy and security standards (2022) and the US Health Insurance Portability and Accountability Act (HIPAA) was enacted in 1996 with the final regulation being published in 2000 (OCR, 2022) to help battle healthcare fraud and to provide regulations governing the privacy and security of an individual's patient details. Both of these policies have evolved over the years to include protected entities and have paved the way to the policies and protective technologies the world sees today aimed at protecting PII.

The tech industry specifically has had to adjust to these policy changes regarding the tracking of individuals, the usage of data from online profiles and platforms, and the right to be forgotten entirely from a service or platform (*Right to Erasure*, 2022). While the shift has provided data protections around the globe, the majority of technology users continue to have little to no control over their personal information with third-party data consumers (Tene & Polonetsky, 2012; Trepte, 2020). From an individual researcher's perspective, understanding if identifiable data types exist in a data set can prevent accidental sharing of such data by allowing its detection in the first place and, in the case of this software package, permit for the results to be publishable by sanitizing the text tokens and provide transparency on the reasons why the token was considered to be PII. From a platform user's perspective, detecting PII ahead of publication and understanding why it is considered PII can prevent an accidental disclosure that can later be used by adversaries. This need is what drives the development of PII-Codex.

The PII-Codex Package

PII-Codex is a Python package built to combine the Information Sensitivity Typology works of Milne et al. (Milne et al., 2016), categorizations and guidelines from the National Institute of Standards and Technology (NIST) (McCallister et al., 2010), Department of Homeland Security (DHS) (*Handbook for Safeguarding Sensitive Personally Identifying Information*, 2012), and the Health Insurance Portability and Accountability Act (HIPAA) (*Health Information Privacy*, 2022). It combines these categories to rate the detection on a scale of 1 to 3, labeling it as Non-Identifiable, Semi-Identifiable, or Identifiable as presented by the risk continuum by Schwartz and Solove (Schwartz & Solove, 2011). The package provides a subset of Milne et al.'s Information Sensitivity Typology as some technologies group entries into a singular category or the detection of the entry may not yet be available.

Built into the package is an analyzer service that leverages Microsoft's Presidio library for PII detection and anonymization (Microsoft, n.d.) as well as the option to use the built-in detection adapters for Microsoft Presidio, Azure Detection Cognitive Skill (Azure, 2022), and AWS Comprehend (Services, 2022) for pre-existing detections. The output of the adapters and the analysis service are analysis objects with a listing of detections, detection frequencies, severities, mean risk scores for each string processed, and summary statistics on the analysis made.

The final outputs do not contain the original texts but provide the sanitized or anonymized texts and where to find the detections, should the end-user require this information. In providing this capability, one can prevent the accidental dissemination of private information in downstream research efforts, an issue commonly discussed in cybersecurity research (Beigi & Liu, 2020; Bélanger & Crossler, 2011; Moura & Serrão, 2019).

Design

PII-Codex is broken down into a series of services, utilities, and adapters. For a majority of cases, end-users may already have used Microsoft Presidio, Azure, AWS Comprehend or some other solution to detect PII in text. To account for these cases, adapters were provided to convert the varying detection results into a common form, DetectionResultItem and DetectionResult objects, which are later used by the Analysis Service and Assessment Service. This usage flow is presented in Figure 1.

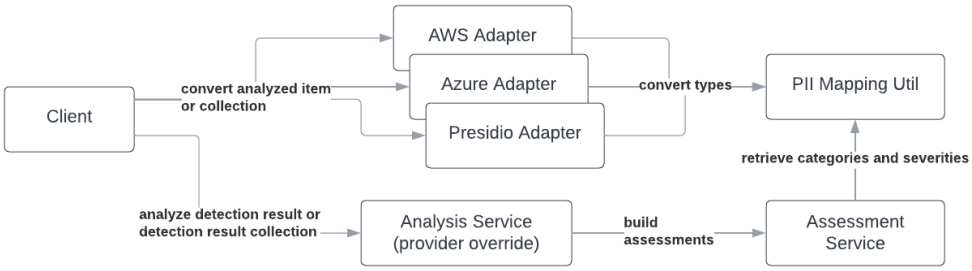

Figure 1: Converting And Analyzing Existing Detections

As shown in Figure 2, for end-users that still require detections to be carried out, Microsoft Presidio was integrated as the primary analysis provider within the Analysis Service.

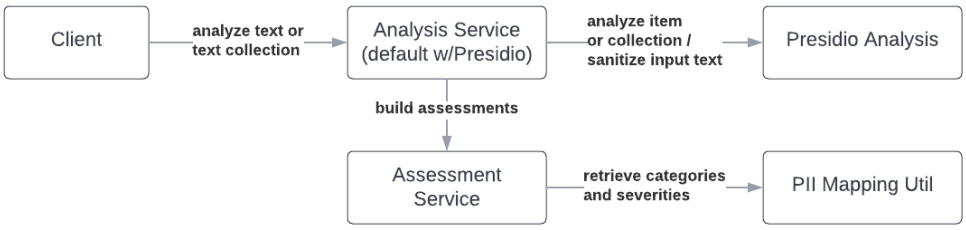

Figure 2: Using Presidio-Enabled Builtin Service for Detection and Analysis

The Analysis and Assessment services expose functions for those defining their own detectors and enable the conversion to a common detection type so that the full Analysis Result set can be built.

Example Usage

The collection analysis permits a list of strings under texts parameter or a DataFrame with a text column under the data parameter. The collection will be analyzed and a summary provided in an AnalysisResultSet object. The AnalysisResultSet object will show individual detections and their risk assessments which includes risk score assessment and associated PII categories. Each analysis is provided with the sanitized input text when using the default analysis service. Unless supplied with another replacement token, the sanitized input text will contain in place of detected PII tokens:

```
Hi! My phone number is <REDACTED>."
```

Email detections, for example, are presented as Identifiable, which automatically places it at a risk level of 3, the highest a token is assigned. Something like a URL is considered Semi-Identifiable and therefore is assigned a risk level of 2. Other texts will fall under Non-Identifiable and will be assigned a risk level of 1.

Using the `texts` parameter:

```
from pii_codex.services.analysis_service import PIIAnalysisService

results = PIIAnalysisService().analyze_collection(
    texts=[
        "email@example.com is the email I can be reached at.",
```

```
            "Their number is 555-555-5555"
        ]
)
```

Using the data parameter with metadata support for social media analysis:

```
import pandas as pd
from pii_codex.services.analysis_service import PIIAnalysisService

results = PIIAnalysisService().analyze_collection(
    data=pd.DataFrame.from_dict({
        "text": [
            "email@example.com is the email I can be reached at.",
            "Their number is 555-555-5555"
        ],
        "metadata": [
            {"location": True, "url": False, "screen_name": True},
            {"location": False, "url": False, "screen_name": True}
        ]
    }),
    collection_name="Social Media Example",
    collection_type="SAMPLE"
)
```

The AnalysisResultSet object will show individual detections and their risk assessments. Email detections, for example, are presented as identifiable and direct PII which automatically place it at a risk level of 3, the highest a token is assigned.

```
{
    "pii_type_detected": "EMAIL_ADDRESS",
    "risk_level": 3,
    "risk_level_definition": "Identifiable",
    "cluster_membership_type": "Personal Preferences",
    "hipaa_category": "Protected Health Information",
    "dhs_category": "Stand Alone PII",
    "nist_category": "Directly PII",
    "entity_type": "EMAIL_ADDRESS",
    "score": 1.0,
    "start": 74,
    "end": 94
}
```

Each string analyzed may contain n number of PII detections, with each detection having a risk severity between 1 and 3 inclusively. The risk score mean $\overline{rs}$ is calculated based on the average of all token risk scores rs for that one string. Since other data is provided, while non-identifiable on its own, may provide context that can lead to identification, their values (assigned a 1 for non-identifiable) are taken into account in the calculation. The calculation for a single string's risk score is presented as the formula below.

$$\overline{rs} = \frac{1}{n}\sum_{i=i}^{n} rs_i \tag{1}$$

For collections of strings being analyzed, each risk score mean is taken into account to provide a collection-wide risk score mean value. Given that a collection can have n number of analyzed strings, the collection risk score mean value can be calculated with the mean of means formula below.

$$\mu_{\overline{rs}} = \frac{\overline{rs}_1 + \overline{rs}_2 + ... + \overline{rs}_n}{n} \tag{2}$$

In the AnalysisResult object, the mean risk score of all detected tokens in a string is provided as the risk score mean. The AnalysisResultSet object will contain the mean of means, or the average of all risk score averages, will be provided as the risk score mean.

Availability

PII-Codex can be installed via pip or poetry. The source code of PII-Codex is available at the GitHub repository (https://github.com/EdyVision/pii-codex). The builds can be obtained from https://github.com/EdyVision/pii-codex/releases and via Zenodo (Rosado, 2023).

References

(2022). In *GDPR.eu*. https://gdpr.eu/what-is-gdpr/

Azure, M. (2022). PII detection cognitive skill - azure cognitive search. In *PII Detection cognitive skill - Azure Cognitive Search | Microsoft Learn*. Microsoft Azure. https://learn.microsoft.com/en-us/azure/search/cognitive-search-skill-pii-detection

Beigi, G., & Liu, H. (2020). A survey on privacy in social media. *ACM/IMS Transactions on Data Science*, *1*(1), 1–38. https://doi.org/10.1145/3343038

Bélanger, F., & Crossler, R. E. (2011). Privacy in the digital age: A review of information privacy research in information systems. *MIS Quarterly*, *35*(4), 1017–1041. http://www.jstor.org/stable/41409971

Handbook for safeguarding sensitive personally identifying information. (2012). Privacy Office, Department of Homeland Security.

Health information privacy. (2022). U.S. Department of Health; Human Services. https://www.hhs.gov/hipaa/index.html

McCallister, E., Grance, T., & Scarfone, K. A. (2010). Guide to protecting the confidentiality of personally identifiable information (PII). *S Department of Commerce: National Institute of Standards and Technology (NIST)*. https://doi.org/10.6028/nist.sp.800-122

Microsoft. (n.d.). *Microsoft/presidio: Context aware, pluggable and customizable data protection and anonymization SDK for text and images*. Microsoft. https://github.com/microsoft/presidio

Milne, G. R., Pettinico, G., Hajjat, F. M., & Markos, E. (2016). Information sensitivity typology: Mapping the degree and type of risk consumers perceive in personal data sharing. *Journal of Consumer Affairs*, *51*(1), 133–161. https://doi.org/10.1111/joca.12111

Moura, J., & Serrão, C. (2019). Security and privacy issues of big data. *Cyber Law, Privacy, and Security*, 375–407. https://doi.org/10.4018/978-1-5225-8897-9.ch019

OCR, O. for C. R. (2022). Summary of the HIPAA privacy rule. In *HHS.gov*. https://www.hhs.gov/hipaa/for-professionals/privacy/laws-regulations/index.html#:~:text=The%20U.S.%20Department%20of%20Health,1996%20(%22HIPAA%22).

Right to erasure. (2022). Information Commissioners Office. https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/individual-rights/right-to-erasure/#:~:text=Under%20Article%2017%20of%20the,be%20created%20in%20the%20future.

Rosado, E. J. (2023). *pii-codex: a Python library for PII detection, categorization, and severity assessment* (Version 0.4.3). https://doi.org/10.5281/zenodo.7212576

Schwartz, P. M., & Solove, D. J. (2011). The PII problem: Privacy and a new concept of personally identifiable information. *New York University Law Review*, *86*, 1814.

Services, A. W. (2022). PII detection cognitive skill - azure cognitive search. In *What is Amazon Comprehend | AWS Comprehend*. Amazon Web Services. https://docs.aws.amazon.com/comprehend/latest/dg/what-is.html

Tene, O., & Polonetsky, J. (2012). Big data for all: Privacy and user control in the age of analytics. *Northwestern Journal of Technology and Intellectual Property*, *11*.

Trepte, S. (2020). The social media privacy model: Privacy and communication in the light of social media affordances. *Communication Theory*, *31*. https://doi.org/10.1093/ct/qtz035

West, S. M. (2017). Data capitalism: Redefining the logics of surveillance and privacy. *Business &Amp;amp; Society*, *58*(1), 20–41. https://doi.org/10.1177/0007650317718185