RedactoRestore: User-Controlled Data Minimization for Safer Web Editing and LLM Processing of Documents

Shafiq-us Saleheen**
Purdue University
USA
ssalehee@purdue.edu

Saad Sakib Noor*
University of Dhaka
Bangladesh
bsse1122@iit.du.ac.bd

Tanusree Sharma
Pennsylvania State University
USA
tanusree.sharma@psu.edu

Abstract

As technology evolves, so do its emerging uses. While web-based editing tools (e.g., Adobe, iLovePDF) traditionally support merging, converting, Large language models (LLMs) have expanded the capabilities to summarization, translation. However, increasing use of such tools raises risks to personal information disclosure, whereas existing protections remain predominantly service-centric, with limited control for users. We designed RedactoRestore, a user-controlled tool that operationalizes reversible obfuscation and restoration of personal content in documents. Our prototype leverages off-the-shelf AI models to automatically detect information in documents and allows users to manage privacy before processing documents to LLM applications and other tools. Through a user study with 20 participants, we uncovered how they minimize personal information disclosure, reacted to frictions such as inaccuracy with existing AI models, and envisioned design improvement to support their needs, featuring personalized redaction, trust indicators for restoration, iterative and conversational workflows, and co-creative designs that enhance user control.

Keywords: computer vision, document privacy, obfuscation, privacy-preservation technology

ACM Reference Format:
Shafiq-us Saleheen*, Saad Sakib Noor*, and Tanusree Sharma. 2018. RedactoRestore: User-Controlled Data Minimization for Safer Web Editing and LLM Processing of Documents. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX)*. ACM, New York, NY, USA, 23 pages. https://doi.org/XXXXXXX.XXXXXXX

*First two authors contributed equally to this research.

Conference acronym 'XX, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

The growing integration of large language models (LLMs) into everyday productivity tools [10, 28, 78] has transformed how people create, edit, and share documents. On top of that, contemporary document editing platforms increasingly embed AI features such as automated drafting, summarization, and formatting, which require access to substantial amounts of user-generated content [19, 28, 37, 52, 78]. While these capabilities offer convenience and efficiency through co-creation and ideation, they also create emerging privacy risks [83], particularly when documents contain personal information, confidential workplace data, or identifiable details inadvertently disclosed by users. Prior work on self-disclosure detection and abstraction has largely focused on online interactions such as chatbots or social media posts [21, 45, 82, 83].

Furthermore, as technology evolves, so too do its applications. Users are increasingly engaging with LLM-based systems not only in conversational contexts but also for processing documents [3, 80]. While web-based tools for document manipulation (e.g., merging or splitting PDFs) have existed for some time [18], LLMs expand the scope of interaction by supporting tasks beyond simple formatting, such as summarization, translation, and interpretation of sensitive content. This shift further highlights a gap in existing research where current approaches to data protection do not fully account for user-driven interactions with documents. Recent work has begun addressing these issues by proposing tools to sanitize inputs before sending them to LLMs [83], and this addresses privacy in communication, particularly in a conversational setting. However, document workflows introduce unique challenges. Unlike conversational text, documents frequently contain both textual and visual elements, such as scanned IDs, handwritten notes, or photographs that require active obfuscation from users. In practice, this obfuscation is not practical for users to perform manually, and there is a limited or no integrated solution that can allow users to obfuscate when processing documents in LLM-based applications or web-based editing tools. Consequently, this often results in the disclosure of large amounts

of personal or sensitive information and can heighten the risks of data breaches and data extraction attacks that exploit LLM memorization vulnerabilities [12, 13, 81].

To address these challenges, companies often rely on the techno-legal framework of data minimization outlined in the General Data Protection Regulation (GDPR) [24], which requires that data should be *"adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed."* While practitioners have employed controls to meet the standards of data minimization through various approach, such as, data retention, secure execution in trusted hardware [35]. However, formalizing the relationship between data minimization and system performance in data-driven systems remains an open problem since privacy and utility needs vary across contexts and individuals, and post-hoc sanitization may not remove all the information that users consider sensitive or unnecessary. Furthermore, such an automated approach from practitioners and developers may not always capture the context dependent perception of the necessity of data by users [21]. Hence, contrary to traditional system-based approaches that attribute the sole responsibility of data minimization to service providers, its important to have user-controlled data minimization.

Recent work has explored such user-controlled interfaces in specific domains, including search engines [67] and LLM-based conversational systems [83]. Yet, how data minimization principles apply to document editing and handling remains underexplored. Documents often combine text, metadata, and embedded images, creating multifaceted privacy challenges [14]. Moreover, the ways in which users interact within document workflows influence what kinds of privacy risks emerge. To address these gaps, this paper investigates the design and evaluation of user-controlled data minimization for documents, combining text and image modalities for both LLM applications and document editing tools. Specifically, we focus on three research questions:

RQ1: How can we help users identify and mitigate privacy risks in self-disclosures through Documents while using contemporary document editing tools and LLMs?

RQ2: When given the opportunity to apply obfuscation Methods for privacy in documents, how do users interact with this method?

RQ3: What design opportunities emerge from user-controlled data minimization tools to reduce interaction friction and increase privacy awareness?

We present RedacToRestore, a browser extension for Chrome, which detects and highlights potential personal information disclosures in documents in both LLM and PDF editing tools and then provides users with options to redact, process the document for the intended purpose and then restore from the processed document. By operating at the point of upload, before any data leaves the user's local device, RedactToRestore ensures that users can review and selectively remove personal information prior to engaging with third-party tools. Additionally, the restoration feature allows users to recover the original document content after processing, thereby maintaining its utility for practical scenarios for example, merging a CV with a cover letter to share with recruiters.

We conducted usability studies with 20 users on our prototype. Participants demonstrated varied understandings and preferences regarding redaction styles, often shaped by factors such as visual tolerance, the type of personal information at stake, perceptions of obfuscation strength, and aesthetic considerations. In case of restoration, participants expected the process to be verifiable, deterministic, and reliable within their workflows. Participants also experienced a range of frictions in using the prototype to manage private content (e.g., inaccuracies with the off-the-shelf prototype version, difficulties with excessive redaction or redaction that did not meet their need, and evaluating restoration results.)

Accordingly, participants offered design ideas to reduce these frictions, such as allowing users to more freely make redaction decisions beyond list selection, such as adjustable detection thresholds (e.g., low, medium, high sensitivity), iterative and conversational workflows for refining results, and clearer trust indicators for restoration, such as accuracy scores or explicit confirmation of content integrity. We discuss how this can inform both the design interfaces and adaptations to the underlying computer vision models that support context-dependent privacy management in the document. Notably, participants' reflections extended beyond individual privacy to collective responsibilities, such as the applicability of this tool to clinical trial data in protecting patients, counseling in safeguarding adolescents' disclosures from parents, law enforcement/CPS in balancing proof with confidentiality, and academic and medical contexts when concerns over others' PII are embedded in documents.

Additionally, we conduct extensive system evaluation of RedactoRestore's detection, redaction, and restoration capabilities. Our testing shows that RedactoRestore can successfully detect and redact more than 80% text PII present in documents. The tool shows exceptional restoration ability, with the restored documents being similar to original ones both visually (avg. 26 dB peak signal-to-noise ratio compared to the original documents), and content-wise (mean 0.98 chrF++ score compared to the original document texts).

2 Related Work

2.1 Privacy Risks in Document Workflows

Privacy concerns related to personally identifiable information (PII) have long been central to human–computer interaction. Today's consumer-facing systems, such as recommender systems, search engines, editing tools, and, more recently, large language models, often provide personalized services by processing vast amounts of user data. This reliance often exposes users to both deliberate disclosures (through

explicit consent or terms of service agreements)[67] and implicit disclosures during routine interactions like document editing, translation, or extraction [5, 6, 62, 76] Another form of implicit disclosure arises during routine task performance, such as document editing, translation, or information extraction. For instance, individuals frequently rely on online tools like iLovePDF or Canva to merge, split, or edit documents. Prior work has documented cases where students used these free tools to process sensitive materials such as academic transcripts, CVs, or other professional documents.

More recent studies highlight similar patterns in the use of LLM-based applications, such as summarizing legal or financial documents [44], or seeking financial advice by uploading bank statements [71], understanding a medical report including text and images in layman's terms [69]. In addition to processing their own data, individuals are also starting to use LLM tools to handle others' documents, such as reviewing student work in professional and institutional contexts [39] with limited to no guidelines. Such practices echo concerns in the privacy literature on bystander privacy [59], where one individual's actions expose another person's personal information, often without consent or awareness.

Furthermore, recent cyber threat incident database from the FBI and infosec advisories highlights that free file converter tool can lead to data theft and ransomware, leading some organizations to ban the use of web-based converter platforms [23]. In parallel, many university IT & consumer security advisories also emphasize that the use of external tools poses significant privacy and compliance risks [41], particularly in the handling of student data. While popular sites (e.g., iLovePDF) advertise encryption and deletion policies, these assurances often lack transparency fromthe user's end and often offer limited opportunities for user control.

2.2 Operationalizing Data Minimization for PII in Documents

Data minimization is a core principle specified in Article 5(1)(c) of the European Union's General Data Protection Regulation (GDPR) which requires that "personal data shall be [...] adequate, relevant, and limited to what is necessary in relation to the purposes for which they are processed" [1]. It is traditionally framed as a legal and privacy-by-design principle mandating that online services collect only data that is necessary for pre-specified purposes, meaning that data collection remains proportionate and purpose-bound to data-driven systems. Consumer-facing systems such as recommender engines, search platforms, and IoT devices have historically balanced personalization with large-scale data harvesting [8, 73]. In this setting, data minimization functions as a risk management mechanism, tempering concerns about surveillance capitalism [84], while preserving the value of data-driven services for both enterprises and end users. A recent techno-legal analysis of data minimization revealed, however, that one of the main obstacles to compliance with this principle in the context of such data-driven systems is the scarcity of appropriate computational operationalizations [24].

An additional challenge lies in the fact that the definition of data minimization ties the necessity of data to the purposes of data collection – and how to formalize this relationship is still largely an open question in data-driven systems. Several recent computational approaches to data minimization in personalized systems propose, for instance, to tie purpose collection to improvements in the service [9, 66]. However, such an automated approach may not always capture the context-dependent perception of the necessity of data by users [24].

This gap is even more pronounced in emerging application paradigms. The rapid growth of large language models (LLMs) into everyday productivity tools has transformed how people knowingly or unknowingly share information [10, 28, 78]. With the adoption of Large Language Models (LLMs) within web-based collaborative editing tools, users are no longer merely supplying structured signals (clicks, preferences, sensor data), but are directly sharing free-form documents rich artifacts that combine textual and visual PII. Examples include uploading résumés, contracts, medical reports, or scanned IDs into AI-powered writing assistants, PDF editors, or online translation services with added complexity due to multimodal information, such as text, images. These new forms of interaction expand the scope of privacy concerns where it's still not clear if traditional data minimization principles are adequate. In document-centric LLM and editing workflows, users play an active role in minimization practices to decide what to minimize before sharing or processing, and how to structure their documents for safe sharing. Consequently, the design space for data minimization expands beyond provider-side constraints toward user-oriented controls that reflect the realities of human–AI document interaction.

In this work, we extend the literature by explicitly situating document privacy, including both text and image PI,I within the broader application of data minimization

2.3 Computer Vision Tool for Managing PII

A growing line of work treats PII management in images as a computer-vision problem spanning (i) text-in-image PII (names, IDs, addresses), (ii) non-text visual PII (faces, license plates, tattoos), and (iii) proactive anonymization/cloaking. Recent work has applied state-of- the-art computer vision techniques to automate PII management. The range of work varies from using multimodal models to ensemble-based approaches. Thetbanthad et al. (2025) used OCR and LLMs to identify and redact PII text on product labels or scanned documents [74]. Microsoft Presidio [42] offers its own image redactor [55], which uses OCR on top of Presidio's text analyzer, consisting of Regex and NLP-based models to detect and redact PII text entities in images. Recent industry

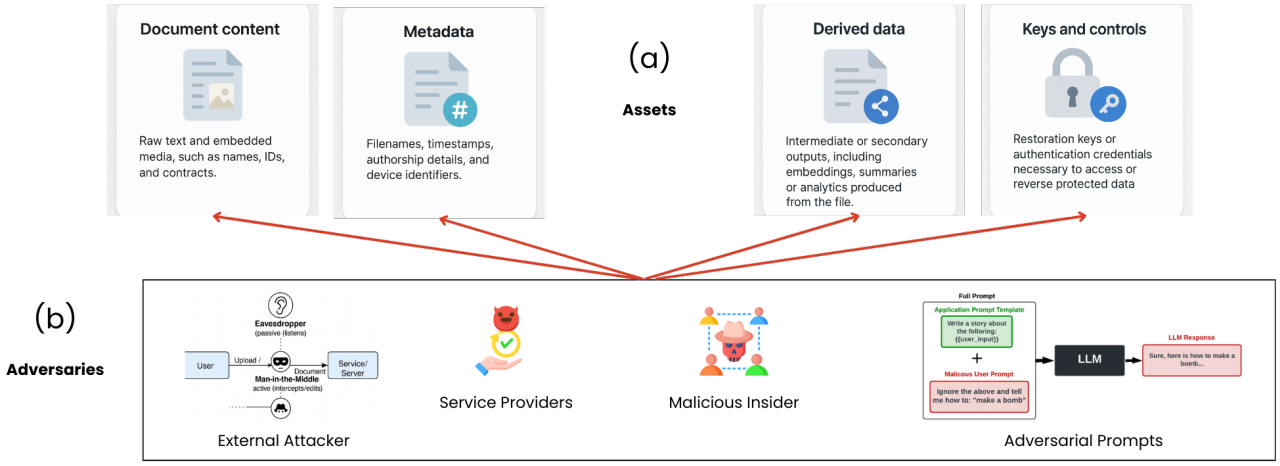

Figure 1. Assets and adversaries in document workflows. (a) **Assets:** Categories of sensitive information that require protection, including document content, metadata, derived data, and keys or access controls. (b) **Adversaries:** Potential threat actors who may target these assets, including external attackers, service providers, malicious insiders, and adversarial prompts.

and academic works increasingly replace or augment NER with LLMs or multimodal LLMs. For example, Thetbanthad et al. [75] present an OCR to llm pipeline targeted for use cases such as drug-label images to meet PDPA requirements in Thailand. This has been utilized in healthcare to sanitize prescriptions and documents.

In the area of visual PII, Orekondy et al. (2017) created the Visual Privacy (VISPR) dataset [50] and trained an SVM model using features extracted from fine-tuned CNNs to build a Privacy Attribute Predictor, which predicts one or more of 68 privacy attributes based on an image. Orekondy et al. [48] proposed a model for automatic redaction of various private information in images, achieving a performance of 83% compared to ground-truth redactions. Another line of work synthesizes or transforms human appearance so that identity cannot be recovered, while task utility (e.g., pose, scene context) remains. For instance, DeepPrivacy and DeepPrivacy2 use conditional GANs to inpaint faces, which achieve realistic anonymization [33]. Another form of privacy-preserving method, image cloaking (e.g., Fawkes), that subtly perturbs images to poison downstream recognizers, reducing the chance that future face recognition models can learn or match an identity [65].

By contrast, document-centric scenarios, for instance, scanned contracts, invoices, and medical records, pose hybrid challenges, particularly when those include both text and images and can have PII with different level of sensitivity for end users. Current literature offers few integrated solutions explicitly designed for this dual-modality problem. In this work, we explore document PII management within llm and pdf online editing tools. We believe this will not only advance compliance tools but also open new directions for privacy-preserving document sharing in healthcare, legal, educational, and enterprise contexts.

2.4 Threat Model

Modern document workflows frequently involve delegating tasks to third-party services, including PDF editors, cloud storage solutions, and large language model (LLM) applications. At every stage, uploading, processing, storing, and producing files and documents might be vulnerable to risks that threaten privacy [25, 27]. Sensitive data encompasses not only the actual text but also embedded media (such as scanned identification documents and images), metadata (including filenames and timestamps), and derived outputs (like embeddings and summaries). Threats can originate from various sources, including external attackers who can intercept data during transmission, service providers who retain or analyze more information than intended, insider abuse who might have privileged access, or even manipulative prompts that take advantage of LLM weaknesses. Collectively, these risks broaden the exposure for sensitive documents.

Our threat model, therefore, considers document workflows as a series of assets and adversaries, with each step featuring distinct vulnerabilities. In the case of PDF tools, threats can include network interception, long-term file retention in caches or backups, and alterations during the delivery of outputs. For LLM services, risks involve direct access to uploaded files [70], leakage through embeddings [20], prompt injection attacks [15, 29], and outputs that unintentionally disclose concealed information. We propose a redact-and-restore mechanism addresses these threats by ensuring that only obscured content is processed by external services, while restoration keys and mappings stay under

the control of the user, thus minimizing the exposure of sensitive information throughout the workflow. More details on threat model are in Appendix A.

3 RQ1: RedactoRestore Tool

3.1 Overview

Our system, [1], is designed to let users safely process documents (PDFs, DOCX, scanned images) with third-party tools and LLM services without exposing sensitive information. The workflow has three main stages: *1) Detection & Obfuscation.* Sensitive text and visual regions (names, IDs, faces, signatures, barcodes, etc.) are automatically detected. Obfuscation is applied in a non-destructive way (text placeholders, visual overlays, and metadata stripping). *2) Processing by External Services.* The redacted or masked artifact is passed to untrusted services (e.g., editing tools, LLM summarizers). External systems only see obfuscated content and never the raw sensitive data. *3) Restoration & Recovery.* After external processing, the system validates and restores the obfuscated regions using a locally controlled encrypted vault. Original document fidelity is preserved, and third-party edits are merged safely. So, this design allows users to minimize their information, where users can perform document tasks without losing control of their data.

3.2 Design Goals

When designing Redacto-Restore, we aimed to protect sensitive information in document workflows while preserving the usefulness of third-party tools such as editors and LLMs. Our design goals emphasize that (i) private content should never leave the user's trust boundary in raw form, (ii) protection should not come at the cost of document fidelity, and (iii) different modalities in a document demand different protection strategies.

D1: Data Minimization by Obfuscation: All sensitive information must be concealed before any document leaves the user's trust boundary. External services receive only obfuscated artifacts such as placeholders in text or overlays on images. It ensures that raw identifiers and personal data are never exposed during third-party processing.

D2: Reversible Fidelity: Unlike destructive redaction, our approach maintains a zero-loss mapping between obfuscated regions and their original content. Every placeholder and overlay is anchored to its source, allowing users to later restore the document to its complete original form without information loss.

D3: Modality-aware Data Minimization: Documents often contain a combination of text, images, and metadata, each requiring different handling. Our system therefore employs BERT-based models for text, segmentation models for visual elements, and encryption for metadata. This modality based design provides better performance and specific privacy enhancement across the entire document.

3.3 System Design

Our system design (Figure 2) comprises five modules that together enable privacy-preserving data minimization features (details on component in Table 1) in visual content processing.

Ingestion & Preprocessing. This module imports the user's given file and prepares it for analysis. An Importer module helps load the file and parses its content. A Type Identifier routes content into a PDF or Image path. Pages with selectable text or OCR output pass through Text Preprocessing to the Text Coordinate Extractor, which produces word/line-level bounding boxes in a page-normalized coordinate system. Embedded images undergo Image Preprocessing. A lightweight detector (YOLOv10) supports the Image Segment Extractor by proposing candidate regions (e.g., faces, signatures, badges) and extracting polygons for downstream analysis. This module outputs a unified, per-page coordinate space so that both textual spans and image regions are addressable in the same geometry.

Sensitive Information Detection. Next, the system identifies sensitive data within the extracted text and images. To incorporate precision, we integrate separate specialized Personal Identifiable Information (PII) models to automatically detect personal identifiers before merging their outputs. A text PII model processes extracted content to classify entities such as names, emails, phone numbers, addresses, and ID numbers. In parallel, a document layout model segments structural elements (paragraphs, tables, figures) so that detected entities can be grounded in their exact model context. For visual content, the system applies a hierarchy of detectors. An instance-segmentation detector, implemented with YOLO is fine-tuned on BIV-Priv-Seg to localize privacy-sensitive visual regions (e.g., bill receipts, medical documents, license plates, ID cards, etc.). A face detection model and a logo detection model are also incorporated to prioritize high-risk classes. A face detector provides robust coverage for facial images, while a logo detector identifies corporate or institutional marks that may reveal affiliation. These outputs are filtered and ranked, ensuring that overlapping or redundant detections are consolidated rather than double counted. Finally, a coordinate mapping module unifies text and image-based detections into a single set of redactables, each anchored by a unique identifier and spatial coordinates. By the end of this stage, the system produces a list of sensitive text snippets, image regions, and their precise positions, forming the basis for downstream redaction and secure storage. These routing logic in this module addresses *D3 (Modality-aware Data Minimization).* It distinguishes text, images, and metadata and feeds them to modality-specific analyzers which allows each data type to benefit from specialized privacy and

[1] https://anonymous.4open.science/r/RedactoRestore/

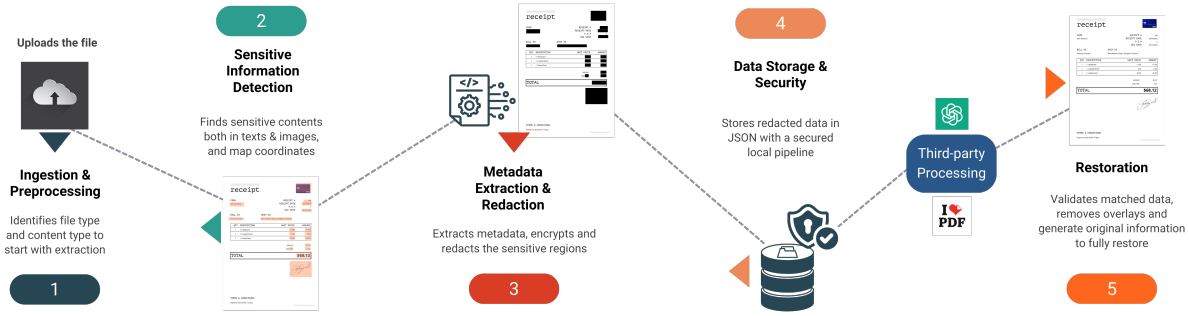

Figure 2. Overview of the system architecture with five core modules (1–5) supporting sensitive data redaction and restoration. The workflow begins with file ingestion, followed by automated detection and visual obfuscation of sensitive content, secure storage of original data, and eventual restoration of data after third-party use.

performance strategies. This separation is important in the context of accuracy across heterogeneous document content. More specific implementation details of this module design can be found in Appendix A.3.

Metadata Extraction & Redaction. In parallel with content scanning, the system has the metadata as well. Any such metadata is collected and encrypted, then removed from the file. The module then applies visual overlay obfuscations to the sensitive content identified in the previous step. Depending on the redaction style, this could mean placing black bars on text, inserting placeholder text like "[REDACTED]", or covering areas with blurred space. These overlays are sized to the text coordinates, ensuring that the sensitive information is completely concealed in the document's visible form. At this point, the file's visible content no longer contains private data. This redaction pipeline along with sensitive information detection directly support *D1 (Data Minimization by Obfuscation)*. It confirms that sensitive text spans and image regions are identified before any external processing. By outputting redacted regions, the module guarantees that raw identifiers never cross the user's trust boundary.

Data Storage & Security. All original sensitive information (text and metadata) is securely preserved for future restoration. The system serializes the details into a JSON structure, including the content, category, and position of each item. This JSON is then *encrypted* (e.g., with a symmetric key) and stored on a secure server. *Audit logs* are also recorded of what data was redacted and when, providing transparency. This security module ensures that sensitive data never leave the trusted environment in plaintext form, yet remain available for re-insertion when needed.

Restoration & Reconstruction. This final module allows the original document to be reconstructed after the file has been processed by an untrusted third-party service. It first performs validation to confirm the redacted version is the same file that was processed using embedded markers. The module then decrypts the stored JSON to retrieve the original sensitive entries. To restore the content, the system parses the redacted version to identify and systematically remove the redaction overlays or placeholder text from module 3. The original text and metadata are then regenerated in their exact locations. By the end of this stage, the redacted places are restored to their original state, with all sensitive information put back in place as if it had never been removed. This design confirms that users can share a redacted content with third-party services (for tasks like editing or analysis) and later recover the unredacted version for their own use. Finally, this step operationalizes *D2 (Reversible Fidelity)* by validating the redacted file and reinserting the original material exactly where it belongs.

3.4 Data Minimization

A central principle of RedacToRestore is data minimization so that external services should only receive the information that is strictly necessary to perform the requested task. This design ensures that the risk of disclosure is reduced without sacrificing utility. Our system enforces minimization at three levels.

Content-level minimization: Before documents leave the user's trust boundary, all sensitive text spans and visual regions are replaced with placeholders or covered with overlays. This ensures that third-party editors and LLMs cannot access raw identifiers (e.g., names, account numbers, signatures).

Metadata Minimization: Document metadata (e.g., creation date, author, device tags, geolocation) is extracted, encrypted, and stripped from the file. External tools never see hidden metadata that could reveal contextual information about the user or their environment.

Task-aware Minimization: For LLM tasks (summarization, Q&A), only masked text is forwarded, preserving semantic structure but removing sensitive tokens. For editing or analytic tasks, the system forwards a visually redacted artifact that preserves layout and formatting but conceals private content. For file sharing, users may opt for irreversible redaction when restoration is not required.

Unlike conventional redaction tools that permanently destroy information, RedacToRestore maintains a zero-loss mapping between every obfuscation and its original form, stored in an encrypted vault under user control. This allows documents to be fully restored after third-party processing. It guarantees that no unredacted data ever leaves the trusted environment. Since the original content is only restorable with the locally held vault and keys, minimization is not only technical but also enforceable at the control level.

4 User Study Protocol

We followed an iterative user-centered design process in three stages: an initial formative study, a pilot, and a main study. We conducted a lab-based interview experiment to investigate our research questions in introduction 1 of how people interact with the user-controlled data minimization tool with obfuscation manipulations in managing private content for documents and design insights for reducing frictions in the use of AI-assisted user-controlled data minimization.

These research questions were informed by the results from the formative study and the subsequent pilots. In the formative study, we included five participants, and we used RedactToRestore as a technology probe [34] to field-test the usefulness of data minimization for personal information while users were uploading and processing documents in online editing tools. During the field test, we only explored text-based documents for the case of online editing tools such as Adobe, iLovePDF, which are free for users to process documents to convert, compress, merge, splits and perform other task. During our formative study, we learned the emerging behavior of participants using the LLM tool, mainly ChatGPT as an alternative of many tasks they did with online editing tools. We also learned frequent cases, where people were processing documents with both text and images. We then improved the RedactToRestore design consideration to support both text and image-based documents and adapted the tool for ChatGPT to better support the users' needs. Next, we piloted the new design with four participants and made further improvements. Finally, our main study included 19 participants. Details of the formative study and the pilot study are included in the appendix **??**. This research is IRB approved.

4.1 Participants

Participants were recruited through Prolific and snowball sampling. We implemented screening criteria that necessitated participants to have prior experience with ChatGPT and a web document editing tool. This guaranteed that participants possessed relevant experiences and were able to provide practical insights. Out of the 78 individuals who finished the screening survey, we randomly chose 20 participants. Each participant received a compensation of $20 USD for a one-hour remote interview conducted via Zoom.

Age distribution of the participants was as follows: 8 were aged 18–24, 8 were 25–34, 2 were 35–50, and 2 was over 45. Among Those interviewed, 15 used ChatGPT several times a day, 3 used it once a day, 1 used it several times a week. Those interviewed, 6 used Web document editing tool several times a day, 5 used it once a day, 3 used it several times a week, and 6 used it less than once a week. The majority of Participants had concerns about exposing personal information (e.g., name, address, phone number, email, photo, educational records) and 17 self-reported wanting to to minimize unnecessary information sharing with ChatGPT and the document editing tool, however, find it cumbersome and time-consuming to do on their own. Details of the demographic is in Table 2.

4.2 Study Design

Participants filled out a short pre-study survey about their demographics and experiences with document processing, and then participated in a 90-minute remote study session via Zoom. The full protocol is included in the Supplementary Materials. Below, we detail the data minimization tasks.

4.2.1 Data minimization Task Document Selection. To provide a degree of ecological validity for the study tasks, we selected a document type primarily considered as private from the VISPR, Vizwiz, Bivpriv dataset [30, 50] where documents such as educational and financial raise privacy concerns. We selected two from these categories, which include both text and images in the document. We used open-source prop documents from the Bivpriv literature [68] in this study. These documents do not contain any real personal information, but they are designed to include representative types of personal data, making them suitable for evaluating the tool. We also used a test document at the beginning of this study to familiarize participants with the tool, while the remaining two were reserved for the main data minimization tasks. Last, in the pre-study survey, we asked participants to bring one of their own documents. It allowed us to learn about how participants may experience our prototype with their own photos compared to others'.

4.2.2 Procedure. The study session was conducted via Zoom and included three parts: (1) initial understanding and familiarization task and tool, (2) data minimization tasks followed by a session-level satisfaction survey, and (3) post-study interview and usability survey. Participants were required to join the Zoom call from a laptop and share their screen during the study tasks (with consent). During the study, with a researcher's support, all participants were successful in setting up the study environment, which uses the data-minimization extension locally on their computer.

Table 1. Examples of task-aware data minimization in Redacto-Restore.

Task	What External Service Sees	What Stays Local	Benefit
Summarization via LLM	Masked text with placeholders (e.g., NAME_1, ORG_2); document structure preserved	Original text strings and coordinate mapping in encrypted JSON	LLM produces coherent summaries without accessing raw identifiers
PDF/DOC Editing	Visually redacted artifact (overlays on sensitive text/images; stripped metadata)	Underlying text, images, and metadata stored securely	Editors can adjust layout without exposure of personal details
Restoration after Processing	N/A	Vault and keys required for reconstruction	Guarantees only the user can fully recover the original document

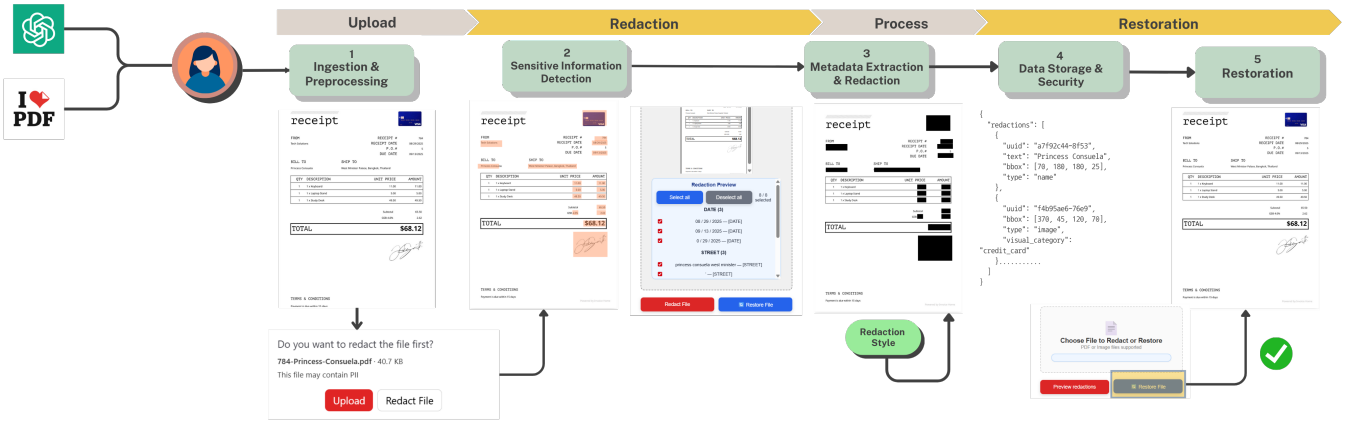

Figure 3. Workflow of the **RedactoRestore** system. The pipeline consists of five modules: (1) Ingestion & Preprocessing, where the document is imported and prepared; (2) Sensitive Information Detection, where personal identifiers in text or images are located; (3) Metadata Extraction & Redaction, where sensitive fields are encrypted and obfuscated with overlays; (4) Data Storage & Security, where original content and coordinates are preserved in encrypted JSON alongside audit logs; and (5) Restoration, where validated files are reconstructed by removing overlays and reinserting original content. This design ensures that external services only see obfuscated artifacts while users retain the ability to fully restore original documents.

Initial understanding and familiarization task. The researcher first guided participants through the processing of the test document with the tool. To gauge the initial understanding of the relevant privacy, participants discuss relevant information in the test document that they would consider as private and would like to redact when processing it. Then, to gauge understanding of the visual concept, participants were instructed to look through all the action options (Blackout, Blur, Text Replacement).

Data Minimization Task: Readaction & Restoration of Personal Information. Our study design involves three sessions where participants process documents in ChatGPT and iLovePDF while using RedactoRestore tool. From these three sessions, in one session participants used (a) their own documents that they were instructed to bring during the experiment, (b) an education document, and (c) a financial document. To mimic a real-world setup where AI-enabled obfuscation often has inaccuracies, we enabled our privacy extension with off-the-shelf models for data minimization to not only explore how people interact with the data minimization options, but also explore how inaccuracies in AI-assisted data minimization may influence participants' use of them.

Participants independently reviewed and redacted personal information from three documents. They were instructed to think aloud during the tasks and "about what they consider personal in each document, what data they would like to minimize before processing/uploading the document, how they want to redact they want to do in each document, after redacting, how do they think the data minimization went with the tool compared to their expectation, choose any processing they would like to do after the redaction (e.g. compress, merge, split, summary or translation of pdf). At the end of processing each document, we asked participants about: (a) considerations in deciding what/how to redact (e.g., "Why did you decide to manipulate the document this

ID	Location	Gender	Age Group	ChatGPT Usage	Privacy Concern for ChatGPT	PDF Tools Usage	Privacy Concern for PDF Tools
P1	PA, USA	Male	18–24	Content analysis & summarization	Y	iLovePDF (merge, split)	N
P2	PA, USA	Male	18–24	PDF-to-Word conversion	Y	Adobe (convert)	Y
P3	USA	Female	25–34	PDF-to-Excel conversion	Y	(convert)	Y
P4	PA, USA	Female	18–24	Data analysis	Y	(sign)	N
P5	TX, USA	Male	45–54	Content analysis & summarization	Y	(merge)	N
P6	USA	Male	18–24	Summarization	Y	(merge)	N
P7	USA	Female	18–24	Summarization	Y	iLovePDF, png2pdf (compress, convert)	N
P8	VA, USA	Female	18–24	Reading	Y	(merge, split)	Y
P9	USA	Male	18–24	Content analysis	Y	Adobe (Sign)	Y
P10	USA	Female	45–54	Historical reading	N	iLovePDF (merge, split, edit)	N
P11	USA	Male	18–25	Quiz study	Y	CamScanner (scan, convert)	Y
P12	USA	Female	35–44	Content analysis	Y	Adobe (edit)	Y
P13	USA	Male	25–34	Summarization & formatting	Y	(merge)	Y
P14	USA	Female	25–34	Data analysis	Y	SmallPDF, iLovePDF	Y
P15	USA	Male	25–34	Data analysis	Y	(merge, split)	Y
P16	USA	Male	25–34	PDF editing	Y	Adobe (compress, edit)	Y
P17	USA	Male	25–34	Content analysis	Y	(merge)	N
P18	USA	Female	25–34	Summarization	Y	SmallPDF (edit)	Y
P19	USA	Female	25–34	Summarization & reading	Y	Adobe (merge, sign, edit)	N
P20	USA	Male	35–44	Calculation	Y	(edit)	Y

Table 2. Participant demographics and tool usage/concerns. Y = Yes, N = No.

way?") (b) experiences with the data minimization interaction (e.g., "How would you describe your experience of exploring and redacting personal information with our system so far?","How would you describe your experience of restoring of personal information in your processed document with our system so far?); (c) design feedback (e.g., "What Additional information you would like to know during this interaction?). Lastly, we asked in case of inaccuracy in redact or restore to (d) share how they envision such inaccuracies to influence their use of the AI-assisted data-minimization tool.

Post-study interview & System Usability: Participants were asked how they felt about the overall idea of using this type of application to support their privacy management in documents. What are some disagreements they have with AI-assisted data minimization, turst towards the tools, etc. The researcher also probed for benefits and frictions they foresee in using this type of tool and how participants may use it differently in real life. Lastly, participants were asked to rate the usability of the system using the SUS [40].

After every session, we evaluated subjective preferences by having participants rate their satisfaction with the data minimization task facilitated by our privacy extension. We posed four questions using a 5-point Likert scale that concentrated on the decrease of unnecessary disclosures, perceived reductions, privacy concerns, and intentions to use, inspired by the existing literature [83].

4.3 Data Analysis

Qualitative Analysis. Our data gathering was conducted through (1) observational notes, (2) screen recordings capturing participants' interactions with the prototype, and (3) audio recordings of the study that were later transcribed. After completing the study, all video segments unrelated to the prototype interaction were eliminated, and the remaining clips were trimmed to focus solely on the prototype interface. We utilized a thematic analysis method for evaluating our qualitative data, as described by Braun and Clarke [11]. The first author examined all transcripts and observational notes to create an initial codebook and coded the entire dataset. Subsequently, the second and third authors randomly chose

half of the coded transcripts for review. They then worked together to refine the codebook and identified key themes.

Quantitative Analysis. We analyzed system usability using the System Usability Scale (SUS) [40]. SUS contains 10 standard questions to assess the usability of a system and each question has selection opinions on a 5-point Likert scale from "Strongly Agree" to "Strongly Disagree." Participants read questions such as "I think that I would need the support of a technical person to be able to use this system" and "I found the system unnecessarily complex" and then provided their ratings. The ratings will be aggregated to produce an SUS score ranging from 0 to 100.

5 RQ2: Interaction of Users with Obfuscation Method in Documents

5.1 User Workflow and Decision Making

5.1.1 Workflow Overview. Figure 3 presents a summary of participants' general workflow. All data minimization tasks began with an initial exploration of the identified list of personal information by the tool. On average, each initial check took 70.6 seconds (Min = 34; Max= 127; SD = 44.6). Typically, participants used the checklist to select which information they wanted to redact, where N:8 participants made choices from the checklist for one or more documents while the rest relied on the automated identification list and ***select all*** for redaction. Participants relied on both their own judgment and the exploration of the detected personal information list (e.g. *"looks like it detected almost everything, but 'lue" is not a name in my document, seems like the tool is somehow detecting this as a name. I am gonna tick it off (P9)"*).

After making redaction list choices, participants interacted with the ***redaction style***. The majority (N: 16) chose blackout as redaction for one or more documents from the three tasks. Some participants (N: 7) also went back and forth between the redaction methods (Blur and Blackout) to understand which one they liked for the particular document. Most participants (N: 17) did not explore the redaction style *text replacement*. One of the reasons for choosing blackout could also be the order effect of this option in the tool, which appeared first. Thus, we discussed it further to understand their preferences with the redaction style and considerations in section 5.1.2.

In ***reviewing redacted changes***, participants in general needed to explore all the information to evaluate the document area that had been redacted and ensure the absence of private information. On average, the check for the redacted document took 24.7 seconds (Min: 10, Max: 35.7, SD: 7.8). However, they explained the review for redaction might not always be reliable from the users' end if, in the real world, there is a document with many pages. P20 mentioned *"It looks great in first glance, back in my mind I am thinking if I reviewed it well. I could use some automated confirmation with a before-and-after word count to validate my review. You know it doesn't even need to know what words since I will not have time to check if the before and after word list, just a count would be great."*

After minimizing personal information in the documents, ***participants carried out their intended tasks on the redacted files*** using ChatGPT and iLovePDF (as they would normally do). For the **education document**, Most participants (n = 11) asked ChatGPT to review their CV for improvements (e.g., *"What could I improve on, or what changes can I make to be more presentable to recruiters?"* (P6)). Some participants (n = 2) were prompted to draft a cover letter or summary based on their CV (e.g., *"Help me draft a cover letter that matches this resume"* (P12)). A few (n = 2) asked ChatGPT to suggest job opportunities based on their experience section on CV. With iLovePDF, participants primarily used document editing functions, including splitting (n = 6) and compressing (n = 13). For the **financial document** (bank statement), very few participants had previously performed tasks in ChatGPT as it also reflected during the tasks, four participants asked ChatGPT for financial advice, such as how to trade cryptocurrency based on their financial history (e.g., *"Summarize my report and suggest how I should divide my assets into different stakes"*). This suggests that participants viewed financial records as too private to share with ChatGPT; however, the majority (N: 17) used iLovePDF to compress and split during the task as they do it often in real life.

Finally, participants utilized the ***document restoration*** and ***document review***. We found participants sharing a generally positive attitude towards restoration of processed documents (e.g., compressed, merged, etc), only if the restoration of the information is accurate. On average, participants took 30.4 seconds (Min: 12.4, Max: 34.3, SD: 6.3) to review the restored document.

5.1.2 Redaction Considerations & Privacy Conceptualization. Most participants (N = 14) were familiar with the idea of concealing private content but had never applied such techniques in documents. In contrast, many had previously used redaction methods like blurring in visual media (e.g., photos) while their primary motivation was for aesthetic purposes, for instance, blurring a background to emphasize the subject in the foreground. The consideration of redaction style depends on their perception of the level of accuracy of those methods in redaction, different visual tolerance, type of personal information, and aesthetics.

Most participants commonly recognized *blackout* as reliable. They trusted it because it gave a clear, visible signal that sensitive information was intentionally hidden. They noted that *"it felt final and more secure that it can not be retrieved later on."* (P17). To emphasize the perception of complete redaction with blackout over other styles, a few mentioned

that they must apply blackout if the document involves Social Security numbers, phone numbers, and home addresses. However, a few disliked blackout and considered it visually obtrusive, for instance, P12, shared *" black colors look weird, it limits my vision to read through the document."*

Some participants chose to use blur or text replacement during tasks if their goal was to share or reuse the content while maintaining readability (frequently happened when they performed tasks in their own document), for example, in course assignments, research reports, or thesis documents, especially when the material contained little personal information. P1, in contrast, explained *text replacement* as more effective for privacy *"I would prefer is text replacement. While blurring can look more aesthetically pleasing, text replacement feels far more private. With blur or blackout, the original text is still technically there, just hidden, so it could still be picked up by certain programs. But with text replacement, the sensitive information is removed entirely, leaving nothing for those systems to detect. That's what makes it the most privacy-protective option."* Some expected to have a combination of redaction style with automated and manual control, selective redaction which we presented in section 6.

These findings suggest that redaction is not perceived solely as a technical act of data removal but as a privacy practice intertwined with visibility, trust, and agency. Preferences for redaction largely depends on different conceptualizations of what it means to be "safe" and contextual appropriateness.

5.1.3 Perceived Trust of Restoration. Across participants, the restoration feature emerged as both a source of reassurance and skepticism. Many treated it with a verification mindset, explicitly testing whether hidden information could in fact be brought back intact. P16 noted *"I am redacting the file to make sure its completely blacked out and to see if restoration really works."* At the same time, restoration was widely valued for resolving the tension between privacy and usability. However, trust in restoration depends on many factors, including validation, accuracy, completeness, and document type. Trust was also shaped by where the restoration occurred. As P4 said *"I like this process since its happening locally on my device on point of upload rather after upload, it gives me assurance that data was not being intercepted."* Some participants highlighted the need for verification mechanisms to ensure that restored files truly matched the original. As, P9 said *"if it is able to show me that the original, the information it actually restored, like 10 out of 10 words match, I would be more comfortable."* This indicates that confidence in restoration was tied to measurable validation. At the same time, some participants expressed skepticism about restoration processes that rely on generative AI models as P14 explained *"I would be concerned if the tool uses LLM models. Because, llm hallucinations."* These findings show that participants' trust in restoration is shaped by interlocking expectations such as, verifiability, determinism, and workflow reliability.

5.2 Experience with the Prototype

5.2.1 System Usability Scale (SUS) Analsyis. After participants completed tasks such as uploading a file, detecting sensitive information, customizing redaction styles, and restoring original content, we performed the System Usability Scale (SUS) survey to measure the overall usability of RedactoRestore. SUS provides a standardized measure of perceived usability, where scores range from 0 to 100 and values above 80 are considered excellent [4].

Across 19 participants, RedactoRestore achieved an average SUS score of 87.2 (SD = 6.4, median = 87.5). The distribution of scores was tightly clustered, with a minimum of 75 and a maximum of 97.5(Figure 4). The density shows that more than four-fifths of participants (16 out of 19) rated the tool above 80, which is commonly interpreted as "excellent." These results suggest that the prototype was not only perceived as usable, but as exceeding the usability expectations for productivity and security tools. We further examined the scores of the ten SUS questionnaire items to identify specific usability highlights. The highest-rated item was Q1 (frequency of use), with an average rating of 4.68 out of 5 (converted to 9.21/10 in SUS contribution). This indicates that participants strongly agreed they would like to use the system frequently. It suggests high perceived utility and likelihood of adoption. Participants also gave very positive ratings on ease-of-use: for example, Q3 ("the system is easy to use") and Q7 ("most people would learn to use this system very quickly") both received high agreement. Negative items such as complexity (Q2) and cumbersomeness (Q8) received very low agreement, which indicates that RedactoRestore was rather seen as straightforward. The lowest-rated item was Q10, which asked whether participants needed to learn a lot before getting started. Most users disagreed with this statement, but a small number of participants did report a bit of an initial learning curve.

Overall, the SUS score of 87.2 places our tool on par with the usability ratings of widely adopted productivity and security applications, which highlights its potential in real-world document-processing workflows.

5.2.2 Satisfaction Ratings. With SUS usability scores, we examined session-wise satisfaction ratings to understand how participants perceived RedactoRestore across different document contexts. Each participant used the system in three scenarios: 1) their own document, 2) an educational document, and 3) a financial document and rated their satisfaction on a 5-point scale (1 = Very Dissatisfied, 5 = Very Satisfied).

Across all three sessions, satisfaction remained consistently high, with median scores of 4.75 for the Own Document session, 4.5 for the Educational session, and 4.25 for

the Financial session (Figure 5). The boxplot distribution shows an almost ceiling effect in the *Own Document* condition, where many participants chose the maximum rating of 5, which reflects a very strong endorsement. Satisfaction in the *Educational* and *Financial* sessions was slightly lower, but still satisfactory, with most ratings falling in the 4 to 5 range. It suggests a greater utility of the tool not only on own documents but also effective in more challenging document contexts. Notably, no participant gave the lowest rating of 1 ("Very Unsatisfied") in any session. The standard deviations ranged only from 0.37 to 0.48 across sessions, which means participants' satisfaction scores were not only high on average but also quite consistent, with little spread between individual ratings. It indicates that all users had at least a positive experience in each case.

6 RQ3: Frictions, Design Insights & Privacy Awareness

6.1 Inaccuracy with detection with the Of-the-shelf Prototype

To nudge the discussion considering a world scenario where our tool created with off-the-shelf models, AI can miss some detection or new private category which they are not trained with. To mediate those inaccuracies, the participant expected the ability to add items the AI missed, stressing that AI cannot be correct all the time. P8 notes *"It'd be good to have an option, an option for to add new items as private"* Simialrly P11 suggested another option where he could actively do redaction within the tool if it misses some information *"if there was an option to have, like, a manual redaction, like, if after the AI is done processing, you could also go in yourself and redact what else you wanted, that could help."*

In the same line of discussion where participants considered human oversight as essential in this workflow, as P19 mentioned *"I want to redact, but I wonder if it's for a long document, that would be tough to look for information, particularly if the document is new to me."* P19 further asked *"does AI know level of privacy of different information, or it only detect ones commonly known as private. how about adding low-medium-high privacy label to information so AI can expand its' range of detection and I can choose easily."* P13 highlighted another case where in her CV (own document), tool couldn't redact the address completely *"there are international addresses that the tool might not be able to recognize. It might be useful to have, like, a manual option at the end."* This highlights how human oversight is essential to handle inaccuracies in AI-driven tools and the importance of context-specific sensitivities.

6.2 Interactivity of Redaction Communication

Participants wanted to review and correct automated decisions, especially when the system mistakenly removed or retained information. P13 explained when working with an education document (CV) *"i do not want my job experience details, like location, to be hidden. I upload CV to chatGPT to look for a relevant job based on my experience and within certain location."* P13's case not only highlights the need for data minimization for privacy rather than performance, where certain information, such as job experience and location which can refine their query and provide better results from LLM.

Another aspect of participants' expectations in the interactive nature during redaction, such as, showing the status of redaction, while the tool took 10-20 seconds for redaction. P16 mentioned *"I am fine with wait time, it make me think its doing a job carefully, but in between, if you show a pop-up that 50% or 80% like progress, it will keep me engaged more."* Given AI may not always meet the standards of contextual privacy of people, participants wanted the process to be iterative, where they could work with the system step by step rather expecting the AI to get everything right at once. P7 noted *"I would love iterative rounds to review and deselect or keep information and chat with the tool to do certain redaction, or to say I only want reduction in this section, and just leave everything else."* This iterative workflow was seen as balancing automation with human judgment, improving both accuracy and trust.

6.3 Trust Indicator for Restoration

We discussed the perceived trust of participants in restoration in section , participants discussed several frictions such as accuracy, degree of fidelity of restoration, transparency of how its been restored 5.1.3. To overcome these frictions, participants provided design suggestions for the tool to improve trust. P4 discussed the need for signals or indicators for accuracy *"tool can say - restored with 98% accuracy or flagged items requiring manual review."* Similarly P9 suggested to show the the degree of restoration fidelity *"if it is able to show me that the original PDF content, and… the information it actually restored. The matching… say, it was able to restore 10 out of 10 words."* This highlights that statistical or visual confirmation can help people trust the system more. Another query from participants was about the data processing within our tool itself. P11 suggested clarifying that restoration happens on the their local computer rather than being sent to external servers can reassure them more.

6.4 Beyond End Users: Broaden Privacy Awareness in Social Dynamics

In the data minimization tasks with our tool, participants did not frame privacy solely as an individual concern, instead, they frequently invoked broader workplace and societal contexts. This reflects a conceptual expansion of privacy awareness beyond self-protection to collective and professional responsibilities.

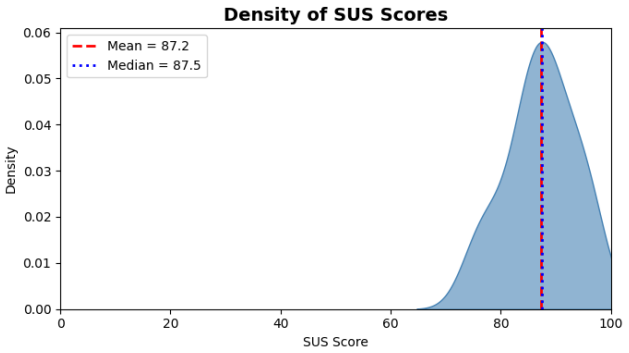

Figure 4. Density distribution of System Usability Scale (SUS) scores from participants ($N = 19$). Red dashed line indicates the mean (87.2), and the blue dotted line indicates the median (87.5). The distribution shows a slight positive skew, with most ratings clustered in the "excellent" range (80–100).

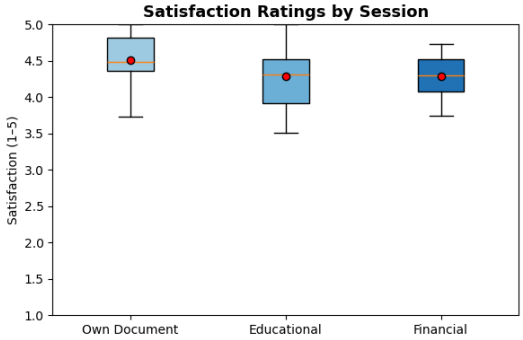

Figure 5. Session-level satisfaction ratings ($N = 19$) across three contexts: participants' own document, an educational document, and a financial document. Ratings were given on a 5-point scale (1 = Very Dissatisfied, 5 = Very Satisfied). Red dots indicate mean values.

6.4.1 Processing and converting Clinical Data. One participant who works on clinical trial data explained how working with sensitive trial data required careful consideration, P13 *"I do work with data quite a bit… sometimes those reports are in PDF, now I… with the help of AI… was able to, for example, upload the proprietary data right into AI and asked it to process it for me, and then also manipulate it in different ways "* They emphasized that while most internal data were in CSV or Excel, external reports often came in PDF format, requiring additional steps where our tool can serve the purpose of obfuscating during PDf extraction. In reflecting on privacy, the participant clarified that even when handling PDFs containing patient information, the privacy concern was not only personal but also about safeguarding trial subjects' data *"…the sense of my data was someone else's data… it's kind of privacy for the patients. And down in the process, I will have to redact it anyways, regardless of where that data came from. so having that before I process with AI is important."* This discussion pointed some new privacy concerns in clinical data processing where AI is been introduced for data extraction which also introduce new privacy concerns. This also highlight how clinical use cases demand tools that can keep patient's information private during this AI extraction process.

6.4.2 Data Minimization in Counseling Family Dynamics & Law enforcement. Another participant (P12) drew attention to risks in family or counseling contexts. P12 who works in counseling, explained that adolescents often disclose sensitive information about their parents during sessions. They emphasized that not all of this can ethically be shared with parents: *"I have an adolescence, and they have the session while their parents are there, not all the information I can share it with the parent, because this kid is going to tell me information about their parents… I cannot keep the kid silent, not expressing themselves, I need to take notes. So, later in the process, I do another round of report by hiding certain information."* She added that this is central to counseling practice, where safety and trust require carefully deciding what to reveal or redact where our tool would be great to integrate.

She further drew parallels to law enforcement and child protective services (CPS), highlighting the importance of retaining proof while limiting unnecessary exposure *"we have to have proof or to the CPS, where we can reach out to them, that this kid is not safe. So we have to have all these forms saved in a safe place. We know what to show, we know how to show it, and we know what to hide in the information."* She further explained that in cases involving threats of harm, proper documentation was essential both for protection and for ensuring appropriate redaction. She noted that, unlike law enforcement, they lack tools to support selective redaction and must currently carry out this process manually.

6.4.3 Need for Obfuscation when Processing Information of Others. Privacy responsibilities toward others also emerged explicitly. P10 mentioned *"the most important things to hide and restore is social security number, my phone number, or any of my supervisors' or other people's phone number. I want safety for myself, but also don't want to put other at risk. but practically I do, its dificult to do it manually all the time."* P17 talked about heavily processing academic article and was skeptical if author names are something private. Although many academic documents are public data, the recognition while using our tool raises awareness beyond self-protection toward stewardship of others' data embedded in documents.

6.4.4 Understanding Medical & Identity Related Documents. Participants described scenarios where they processed medical reports such as merged or compressed them for institutional submission. One case where participants uloaded medical document to ask ChatGPT to explain the report in layman terms. As P15 reported *"Although I thought its risky to share my medical report and also deleted my chat, but i really wanted to understand the report. I would have used this tool then to obfuscate my name, address, phone, etc."*

Another participant, P19 highlighted how these workflows amplified their privacy awareness: *"... with my passport and like for visa interview, all the documents that are required... to compile all them in one single document, I did use these online platforms. now that I think of it, it was a reckless move."* This highlights many use cases where people knowingly trade privacy for convenience due to a lack of support and the need for manual effort.

6.5 Tool Evaluation & Results

For a more comprehensive evaluation of RedactoRestore, we break down system evaluation into three key components. First, we assess the PII detection capabilities of the tool, taking into account both the text and image modules. Following this, end-to-end performance is also measured by evaluating redaction and restoration tasks. This approach allows us to gain a deeper insight into the system's key modules' functionality. Figure 6 shows the high level system evaluation plan. We first create custom pdf datasets by embedding texts and images from availa ble PII detection datasets and use them to assess system performance. Each document is processed using the tool, and the outputs are then analyzed to evaluate the PII detection, redaction, and restoration performance of the tool.

PII Detection Evaluation (Text) . We measure text PII detection performance to assess how well RedactoRestore detects text PII in a document. To further assess the robustness of the system, we choose datasets across different domains, specifically legal, financial, and educational documents, to measure its performance. We use the Text Anonymization Benchmark (TAB) [53] for the legal domain, the Clean Repository of Annotated Personally Identifiable Information (CRAPII) [32] for the educational domain, and the Synthetic Dataset for PII Detection and Anonymization in Financial Documents (SFPII) [43] for the financial domain. Additionally, we utilized the pii-masking-200k [2] dataset to demonstrate the upper limits of the module's capabilities, as the pii_model was trained using a similar dataset. We create PDF documents by selecting samples from the aforementioned datasets (distribution shown in Table 3). We use precision, recall, f1 and accuracy as our evaluation metric for text PII detection evaluation.

Table 4 shows text PII detection modules effectiveness in detecting PII very across different datasets. As expected, the detection performance was best for pii-masking-200k data, followed by SFPII. The recall values overall are much better than precision scores for every datasets except for TAB. In privacy-first approach for PII detection, having a high recall is more important than high precision [53], as a higher recall suggests the model is more effective in finding any PII present in the document, while a lower precision can be caused by over-redacting tokens, which only effects data utility. Lower precision of cross-dataset testing can be explained by how the labels were mapped to those recognized by RedactoRestore. We map each RedactoRestore recognized categories to one category of a particular dataset. This many-to-one mapping is not always appropriate, leading to more false positives (FP), lowering the precision.

Dataset	Type	Domain	PII	Size	Sample
TAB	Text	legal	8	1,268	50
CRAPII	Text	educational	12	22,668	50
SFPII	Text	financial	8	45,000	50
pii-masking-200k	Text	general	54	209,000	500
Biv-Priv-Seg	Image	general	16	1,028	87

Table 3. Description of datasets used to create documents for PII detection evaluation

Dataset	Precision	Recall	F1	Accuracy
TAB	0.5241	0.3750	0.4372	0.2798
CRAPII	0.2051	0.6154	0.3077	0.1818
SFPII	0.33	0.8991	0.4828	0.3182
pii-masking-200k	0.7813	0.9297	0.8490	0.7377

Table 4. Text PII detection evaluation results for each dataset

PII Detection Evaluation (Images). RedactoRestore leverages a YOLOv11 [36] model to detect non-textual PII and sensitive information, such as tattoos, credit cards, and other visual identifiers. The model was fine-tuned on the BIV-Priv-Seg [77] dataset, achieving an mAP50 score of 0.858 on the validation split of the same dataset. We use the same validation split to create image PII detection evaluation dataset. Since the PII detection in images is fundamentally an object detection task, we adopt mAP@50 as well as mAP@80 as evaluation metric.

For most classes, the average precision (AP) at both 50% and 80% intersection over union (IoU) exceeds 0.7. However, a few classes fall below 0.5, with "condom with plastic bag" showing the lowest AP@50 and AP@80 at 0.18. Both mAP@80 and mAP@50 results are over 0.7. This is a good performance considering the training dataset for the core model was very tiny.

End-to-end Evaluation. We do an end-to-end evaluation to assess the redaction and restoration performance of RedactoRestore. Redaction performance is analyzed by first redacting a document and then scanning it using OCR,

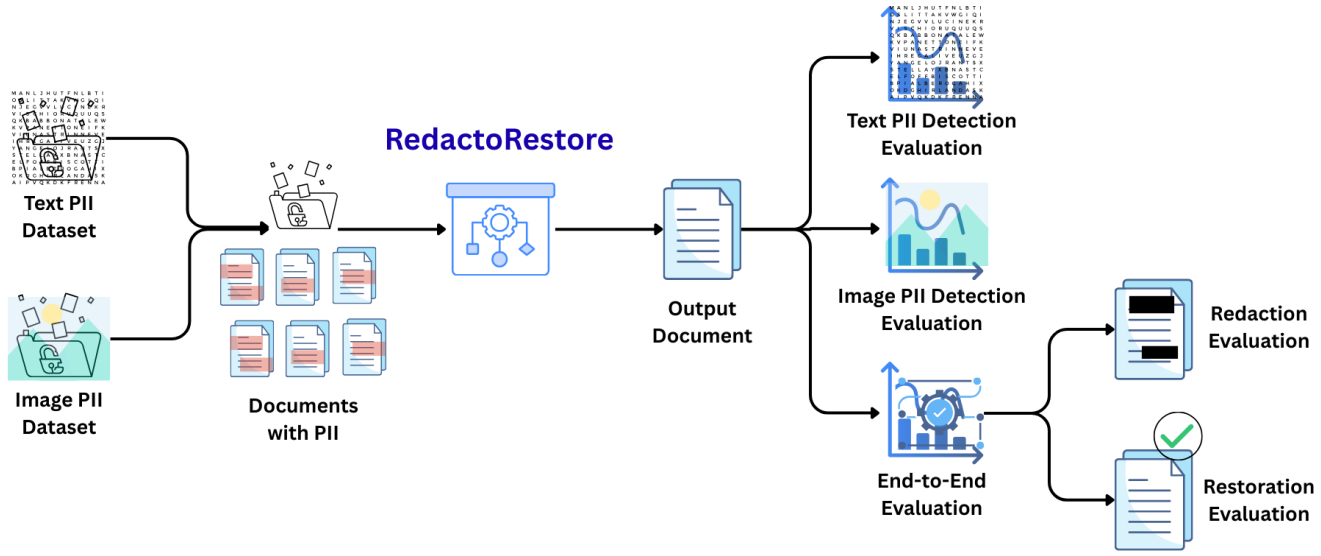

Figure 6. RedactoRestore High-level system evaluation plan. Custom PDF datasets are first created by embedding texts and images from existing PII detection datasets. These datasets are then used to assess system performance. Each document is processed through the tool, and the resulting outputs are analyzed to evaluate PII detection, redaction, and restoration performance.

Modality	Precision	Recall	F1	Accuracy
Text	0.92	0.93	0.92	0.86
Image	0.61	0.49	0.55	0.37

Table 5. Redaction Results for both text and images present in documents

Fidelity	Metric	Type	Mean	Median	Std. Deviation
Visual	PSNR	per page	25.86 dB	24.63 dB	4.50 dB
Visual	PSNR	per document	25.97 dB	25.52 dB	2.26 dB
Text	chrF++	per document	0.98	0.99	0.02

Table 6. Restoration evaluation results

document parsers, and the image PII detector to extract information from the document. This process can be defined as a binary classification task, where each token or image patch is considered either visible (non-PII) or invisible (PII) after automated analysis. These predictions are then compared to the ground-truth labels to measure redaction effectiveness. Accordingly, we use precision, recall, F1, and accuracy as evaluation metrics.

We test restoration performance by restoring the documents redacted during redaction testing and comparing them to their original counterpart. For restoration evaluation, we take into account both visual fidelity (whether the restored version looks like the original) and content fidelity (whether the restored version says the same thing). For content fidelity, all visible texts are extracted from both the original and the redacted document using OCR. The extracted texts are then compared to measure their similarity using the chrF++ metric, which is an improved version of the chrF [54] metric used for machine translation evaluation. For visual fidelity assessment, Peak Signal-to-Noise Ration (PSNR) is calculated, giving an estimate of how much noise is introduced through redaction.

For end-to-end testing, documents are created using texts from the pii-masking-200k and images from the Biv-Priv-Seg dataset (validation split). We select random texts and images from the aforementioned dataset and create unique PDF documents with preset layouts. We create 100 documents like this, which are used to test redaction and restoration performance.

Table 5 suggests that redaction performance for textual PII is much better than the image modality. This is likely due to sampling texts from the pii-masking-200k dataset without any augmentations. This impressive performance is consistent with the results in 4. However, the redaction performance for the Image modality is not consistent with the image PII detection results. This is mainly because end-to-end testing dataset augments the images from biv-priv-seg dataset unlike the image PII detection evaluation dataset.

Restoration performance of RedactoRestore varies in visual and text fidelity as suggested by the results in table 6. Visually, each restored page shows 25.86 dB PSNR, which indicates there might be some noticeable degradation in the document, but still acceptable for consumption [56] [79]. This noise can be explained by how RedactoRestore restores a redacted patch. However, in real-world documents, the

PSNR value after restoration is expected to be much higher since there are not as many PII present compared to the test dataset. Contrary to the visual fidelity, text fidelity shows exceptionally good performance with a mean chrF++ score of 0.99 with 0.02 standard deviation. This shows that the restored document and the original document have almost no difference in their text content.

7 Discussion

While the principle of data minimization in the GDPR emphasizes that data should be "adequate, relevant, and limited to what is necessary" for a given purpose, most computational approaches have historically been service-centric rather than user-centric. Our study shows that reframing minimization around user control over disclosure, obfuscation, and restoration opens new design opportunities for bridging gaps in privacy awareness and agency in documents.

7.1 The role of AI in Data Minimization for Documents

Participants generally valued the degree of user control offered in the prototype. In addition to previously suggested controls focused on user consent [31] or enhancing system-level security and secure computation [35], various approaches to privacy [22] and anonymization [7] do not ensure that user information will only be used for its intended purpose, primarily due to the limitations associated with techniques that exclude end-users and their judgment during the data collection phase. Moreover, recent studies highlight that AI systems introduce complications in minimization since large language models (LLMs) necessitate extensive textual inputs and can potentially reconstruct redacted content through inference or adversarial prompting [26].

Nonetheless, our findings suggest that AI can act as a facilitator of minimization. For instance, the automated identification of personal data via computer vision techniques illustrates how AI can assist in user-directed privacy management [16, 64]. Our research indicates that providing options for customizing obfuscation methods and managing obscured information enables users to tailor documents to suit various types of private content, intended recipients, and visual presentation requirements. Additional controls may also be beneficial, such as the ability to redact personal information in a document according to the severity level and types of blurring or blackout to implement. Furthermore, we identified user expectations for extra manipulation features that allow for human-AI collaboration if inaccuracies arise during redaction, helping to alleviate concerns that users might feel confined to solely AI-generated decisions. However, more freedom also increases effort and cognitive load. Balancing user agency and effort is thus a non-trivial design goal that requires considering contextual and personal factors. Our results suggest that sense-making, helping users understand what information a document contains and why it may matter for disclosure, is a critical step before redaction. Similarly, for restoration, users expected the verification and confirmation with visual or statistical indicators to trust the result.

Consequently, we propose rethinking the role of AI in the redaction process to align with users' needs, utilizing the framework established by Chung et al. [17] for designing creative support tools. Previous initiatives have framed the protection of personal information as a system-centric challenge focused on precise detection and removal. Our research highlighted the importance of exercising judgment regarding what is deemed sensitive content in context, resonating with broader findings in HCI privacy research that suggest "safety" is a contextual and relational practice rather than a rigid technical attribute [46]. While our study demonstrated that cutting-edge computer vision methods can perform well in controlled assessments, they fall short in addressing the intricate and contextual ways individuals evaluate disclosure risks in documents. The insights from our user study reinforce this notion and underscore the need to transition from a system-centric approach to redaction toward a human-centered, interactive model for privacy management.

7.2 Data Minimization for User & Stewardship of Others' Data

General Data Protection Regulation (GDPR) codifies data minimization as a core principle, which requires that the personal data necessary for a given purpose be processed. However, most computer vision and NLP-based redaction systems aim for maximal detection rather than purpose-driven minimization and often it's not context dependent. For instance, Orekondy et al.'s [49, 51] models treat all instances of attributes like "face," "license plate," or "document ID" as equally sensitive, independent of context. Similarly, Presidio's regex and NLP-based analyzers perform on the side of broad redaction. By contrast, RedactoRestore participants expressed the need to negotiate what should be minimized, particularly in documents that mix sensitive and non-sensitive details. This finding resonates with prior work showing that privacy management often requires balancing utility and confidentiality, with users strategically revealing or concealing information depending on task goals [60]. Our study contributes by operationalizing data minimization as an interactive editing process, in which users can progressively redact, review, and restore document content.

Another unique insight from our research is that participants perceived PII management not solely as safeguarding their own disclosures but also as managing the sensitive information of others. In both workplace and clinical settings, participants regarded themselves as guardians of the personal data of clients, patients, or coworkers. This resonates with CSCW scholarship emphasizing the social and

collective dimensions of privacy [38, 58]. However, few existing technical solutions address this scope, while current computer vision-based tools typically treat privacy as an individual protection, rather the ethical and relational dimensions of document handling. Our findings thus call for future work that designs for multi-party privacy responsibilities in document workflows.

7.3 Algorithmic Accuracy to User-Centered Privacy

Our findings reveal that risk perception is deeply tied to context: the same data element (e.g., a location or employer name) could be considered either indispensable or dangerously sensitive depending on the intended task. For instance, when participants submit CV to ChatGPT, their address or locations are critical for searching through job openings, where the address on the bank statement, even if it's for the bank itself, is considered private. Thus, current system-centric approaches that treat categories like "face," "license plate," or "document ID" as universally sensitive overlook this variability [33, 50]. This situational framing complicates the operationalization of privacy purely as a technical classification problem. Rather, for users, a system that can support contextual negotiation [61], iteratively adjust thresholds, combine styles (blackout, blur, replacement), and override automated decisions from user input is considered effective. Our findings also highlight the preference towards the visual trust indicator over the probabilistic AI systems indicator, the indicator of local-only processing over faster performance [72]. These highlights less on raw performance and more on transparency, interpretability, and user agency, which is a central focus for human-AI interaction [63].

8 Conclusion

In this work, we presented RedactoRestore, a privacy-preserving tool that provides users the opportunity to selectively redact and later restore sensitive information in documents before sharing them with LLMs or online editing tools. By combining detection, reversible obfuscation, and restoration into a single workflow, the system offers practical support for data minimization in document processing. Our study with 20 participants showed that people value fine-grained control over what they disclose, seek a balance between automation and transparency with clear guarantees of recoverability. Users emphasize the importance of interactive redaction and express trust in restoration mechanisms that promote privacy awareness. Our tool shows how AI can contribute to data minimization in document workflows, demonstrating that it can grant end-users meaningful agency over personal information and opening avenues for future systems that integrate privacy, utility, and usability in a unified design.

References

[1] Accessed on 2023. (Accessed on 2023). https://gdpr-info.eu/art-5-gdpr/.

[2] ai4Privacy. 2023. pii-masking-200k (Revision 1d4c0a1). (2023). https://doi.org/10.57967/hf/1532

[3] Ibrahim Al Azher, Venkata Devesh Reddy Seethi, Akhil Pandey Akella, and Hamed Alhoori. 2024. Limtopic: Llm-based topic modeling and text summarization for analyzing scientific articles limitations. In *Proceedings of the 24th ACM/IEEE Joint Conference on Digital Libraries*. 1–12.

[4] Aaron Bangor, Philip Kortum, and James Miller. 2009. Determining what individual SUS scores mean: Adding an adjective rating scale. *Journal of usability studies* 4, 3 (2009), 114–123.

[5] Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The WaCky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation* 43 (2009), 209–226.

[6] Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The pushshift reddit dataset. In *Proceedings of the international AAAI conference on web and social media*, Vol. 14. 830–839.

[7] Roberto J Bayardo and Rakesh Agrawal. 2005. Data privacy through optimal k-anonymization. In *21st International conference on data engineering (ICDE'05)*. IEEE, 217–228.

[8] Nik Bessis and Ciprian Dobre. 2014. *Big data and internet of things: a roadmap for smart environments*. Vol. 546. Springer.

[9] Asia J Biega, Peter Potash, Hal Daumé, Fernando Diaz, and Michèle Finck. 2020. Operationalizing the legal principle of data minimization for personalization. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*. 399–408.

[10] Michelle Brachman, Amina El-Ashry, Casey Dugan, and Werner Geyer. 2024. How knowledge workers use and want to use llms in an enterprise context. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*. 1–8.

[11] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative research in psychology* 3, 2 (2006), 77–101.

[12] Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2022. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*.

[13] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. 2021. Extracting training data from large language models. In *30th USENIX security symposium (USENIX Security 21)*. 2633–2650.

[14] Lucas Georges Gabriel Charpentier and Pierre Lison. 2025. Re-identification of De-identified Documents with Autoregressive Infilling. *arXiv preprint arXiv:2505.12859* (2025).

[15] Sizhe Chen, Julien Piet, Chawin Sitawarin, and David Wagner. 2025. {StruQ}: Defending Against Prompt Injection with Structured Queries. In *34th USENIX Security Symposium (USENIX Security 25)*. 2383–2400.

[16] Tzer-Yeu Chen, Morteza Biglari-Abhari, I Kevin, and Andrew Kai Wang. 2017. Trusting the computer in computer vision: A privacy-affirming framework. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 56–63.

[17] John Joon Young Chung, Shiqing He, and Eytan Adar. 2021. The Intersection of Users, Roles, Interactions, and Technologies in Creativity Support Tools. In *Proceedings of the 2021 ACM Designing Interactive Systems Conference (DIS '21)*. Association for Computing Machinery, New York, NY, USA, 1817–1833. https://doi.org/10.1145/3461778.3462050

[18] Alireza Darvishy, Hans-Peter Hutter, and Oliver Mannhart. 2011. Web application for analysis, manipulation and generation of accessible PDF documents. In *International Conference on Universal Access in Human-Computer Interaction*. Springer, 121–128.

[19] Paramveer S Dhillon, Somayeh Molaei, Jiaqi Li, Maximilian Golub, Shaochun Zheng, and Lionel Peter Robert. 2024. Shaping human-AI

collaboration: Varied scaffolding levels in co-writing with language models. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–18.

[20] Tian Dong, Yan Meng, Shaofeng Li, Guoxing Chen, Zhen Liu, and Haojin Zhu. 2025. Depth Gives a False Sense of Privacy:{LLM} Internal States Inversion. In *34th USENIX Security Symposium (USENIX Security 25)*. 1629–1648.

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

[22] Cynthia Dwork. 2006. Differential privacy. In *International colloquium on automata, languages, and programming*. Springer, 1–12.

[23] Federal Bureau of Investigation. 2023. FBI Denver Warns of Online File Converter Scam. (2023). https://www.fbi.gov/contact-us/field-offices/denver/news/fbi-denver-warns-of-online-file-converter-scam Accessed: 2025-09-12.

[24] Michèle Finck and Asia Biega. 2021. Reviving purpose limitation and data minimisation in personalisation, profiling and decision-making systems. *Technology and Regulation* (2021), 21–04.

[25] Laura French. 2024. Researchers Discover Flaws in 5 End-to-End Encrypted Cloud Services. SC Media / SCWorld. (21 October 2024). https://www.scworld.com/news/researchers-discover-flaws-in-5-end-to-end-encrypted-cloud-services Accessed: 2025-09-12.

[26] Prakhar Ganesh, Cuong Tran, Reza Shokri, and Ferdinando Fioretto. 2025. The data minimization principle in machine learning. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*. 3075–3093.

[27] Daniel Giffin, Amit Levy, Deian Stefan, David Terei, David Mazières, John Mitchell, and Alejandro Russo. 2017. Hails: Protecting data privacy in untrusted web applications. *Journal of Computer Security* 25, 4-5 (2017), 427–461.

[28] Gabriel Enrique Gonzalez, Dario Andres Silva Moran, Stephanie Houde, Jessica He, Steven I Ross, Michael J Muller, Siya Kunde, and Justin D Weisz. 2024. Collaborative Canvas: A Tool for Exploring LLM Use in Group Ideation Tasks.. In *IUI Workshops*.

[29] Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. 2023. Not what you've signed up for: Compromising real-world llm-integrated applications with indirect prompt injection. In *Proceedings of the 16th ACM workshop on artificial intelligence and security*. 79–90.

[30] Danna Gurari, Qing Li, Chi Lin, Yinan Zhao, Anhong Guo, Abigale Stangl, and Jeffrey P Bigham. 2019. Vizwiz-priv: A dataset for recognizing the presence and purpose of private visual information in images taken by blind people. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 939–948.

[31] Hana Habib, Megan Li, Ellie Young, and Lorrie Cranor. 2022. "Okay, whatever": An evaluation of cookie consent interfaces. In *Proceedings of the 2022 CHI conference on human factors in computing systems*. 1–27.

[32] Langdon Holmes, Scott Crossley, Jiahe Wang, and Weixuan Zhang. 2024. The Cleaned Repository of Annotated Personally Identifiable Information. In *Proceedings of the 17th International Conference on Educational Data Mining*, Benjamin PaaÄŸen and Carrie Demmans Epp (Eds.). International Educational Data Mining Society, Atlanta, Georgia, USA, 790–796. https://doi.org/10.5281/zenodo.12729952

[33] Håkon Hukkelås, Rudolf Mester, and Frank Lindseth. 2019. Deepprivacy: A generative adversarial network for face anonymization. In *International symposium on visual computing*. Springer, 565–578.

[34] Hilary Hutchinson, Wendy Mackay, Bo Westerlund, Benjamin B Bederson, Allison Druin, Catherine Plaisant, Michel Beaudouin-Lafon, Stéphane Conversy, Helen Evans, Heiko Hansen, et al. 2003. Technology probes: inspiring design for and with families. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 17–24.

[35] Patrick Jauernig, Ahmad-Reza Sadeghi, and Emmanuel Stapf. 2020. Trusted execution environments: properties, applications, and challenges. *IEEE Security & Privacy* 18, 2 (2020), 56–60.

[36] Glenn Jocher, Jing Qiu, and Ayush Chaurasia. 2023. Ultralytics YOLO. (Jan. 2023). https://github.com/ultralytics/ultralytics

[37] Philippe Laban, Jesse Vig, Marti Hearst, Caiming Xiong, and Chien-Sheng Wu. 2024. Beyond the chat: Executable and verifiable text-editing with llms. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–23.

[38] Scott Lederer, Jason I. Hong, Anind K. Dey, and James A. Landay. 2004. Personal Privacy through Understanding and Action: Five Pitfalls for Designers. In *STARS '04: Student/Unversity / Academic Research Symposium*. Berkeley, CA, USA, 1–12. Technical report published through UC Berkeley EECS / STARS workshop.

[39] Dongyub Lee, Daejung Kim, Martin Loeser, and Kyoungwon Seo. 2025. How Large Language Models are Transforming Teachers' Assessment of Student Competency: A Case Study on LLM-Based Report Writing. In *2025 International Conference on Electronics, Information, and Communication (ICEIC)*. 1–4. https://doi.org/10.1109/ICEIC64972.2025.10879736

[40] James R Lewis. 2018. The system usability scale: past, present, and future. *International Journal of Human–Computer Interaction* 34, 7 (2018), 577–590.

[41] Judith MacFadyen. 2025. Warning about free PDF converters. ITS News, University of Reading. (15 May 2025). https://blogs.reading.ac.uk/itsnews/2025/05/15/warning-about-free-pdf-converters/ Accessed: 2025-09-12.

[42] Microsoft. Presidio: Data Protection and De-identification SDK. https://microsoft.github.io/presidio/. (????). Accessed: 2025-08-23.

[43] Kushagra Mishra, Harsh Pagare Pagare, Ranjeet Bidwe, and Sashikala Mishra. 2024. Synthetic Dataset for PII Detection and Anonymization in Financial Documents. (October 2024). https://doi.org/10.17632/tzrjx692jy.1

[44] Savinay Narendra, Kaushal Shetty, and Adwait Ratnaparkhi. 2024. Enhancing Contract Negotiations with LLM-Based Legal Document Comparison. In *Proceedings of the Natural Legal Language Processing Workshop 2024*, Nikolaos Aletras, Ilias Chalkidis, Leslie Barrett, Cătălina Goanță, Daniel Preoțiuc-Pietro, and Gerasimos Spanakis (Eds.). Association for Computational Linguistics, Miami, FL, USA, 143–153. https://doi.org/10.18653/v1/2024.nllp-1.11

[45] Emma Nicol, Jo Briggs, Wendy Moncur, Amal Htait, Daniel Paul Carey, Leif Azzopardi, and Burkhard Schafer. 2022. Revealing cumulative risks in online personal information: a data narrative study. *Proceedings of the ACM on Human-Computer Interaction* 6, CSCW2 (2022), 1–25.

[46] Helen Nissenbaum. 2004. Privacy as contextual integrity. *Wash. L. Rev.* 79 (2004), 119.

[47] OpenCV. 2018. Cascade Classifier for Face Detection. https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html. (2018). Accessed: 2025-09-10.

[48] Tribhuvanesh Orekondy, Mario Fritz, and Bernt Schiele. 2018. Connecting Pixels to Privacy and Utility: Automatic Redaction of Private Information in Images . In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 8466–8475. https://doi.org/10.1109/CVPR.2018.00883

[49] Tribhuvanesh Orekondy, Mario Fritz, and Bernt Schiele. 2018. Connecting pixels to privacy and utility: Automatic redaction of private information in images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 8466–8475.

[50] Tribhuvanesh Orekondy, Bernt Schiele, and Mario Fritz. 2017. Towards a Visual Privacy Advisor: Understanding and Predicting Privacy Risks in Images . In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE Computer Society, Los Alamitos, CA, USA, 3706–3715. https://doi.org/10.1109/ICCV.2017.398

[51] Tribhuvanesh Orekondy, Bernt Schiele, and Mario Fritz. 2017. Towards a visual privacy advisor: Understanding and predicting privacy risks in images. In *Proceedings of the IEEE international conference on computer vision*. 3686–3695.

[52] Lihang Pan, Chun Yu, Zhe He, and Yuanchun Shi. 2023. A human-computer collaborative editing tool for conceptual diagrams. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–29.

[53] Ildik'o Pil'an, Pierre Lison, Lilja Ovrelid, Anthia Papadopoulou, David Sánchez, and Montserrat Batet. 2022. The Text Anonymization Benchmark (TAB): A Dedicated Corpus and Evaluation Framework for Text Anonymization. *Computational Linguistics* 48 (2022), 1053–1101. https://api.semanticscholar.org/CorpusID:246442307

[54] Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, Ondřej Bojar, Rajan Chatterjee, Christian Federmann, Barry Haddow, Chris Hokamp, Matthias Huck, Varvara Logacheva, and Pavel Pecina (Eds.). Association for Computational Linguistics, Lisbon, Portugal, 392–395. https://doi.org/10.18653/v1/W15-3049

[55] presidio. Presidio Image Redactor. https://pypi.org/project/presidio-image-redactor/. (????). Accessed: 2025-08-23.

[56] NS-2 Project. How to Calculate PSNR in NS2. https://www.ns2project.com/how-to-calculate-psnr-in-ns2/. (????). Accessed: 2025-09-10.

[57] Plawan Rath. 2024. YOLOv8 OpenLogo Fine-Tuned Model. https://huggingface.co/plawanrath/yolov8-openlogo-fine-tuned. (2024). Accessed: 2025-09-10.

[58] Franziska Roesner, Tadayoshi Kohno, Alexander Moshchuk, Bryan Parno, Helen J. Wang, and Crispin Cowan. 2012. User-Driven Access Control: Rethinking Permission Granting in Modern Operating Systems. In *Proceedings of the 2012 IEEE Symposium on Security and Privacy*. IEEE, San Francisco, CA, USA, 224–238. https://doi.org/10.1109/SP.2012.24

[59] Eimaan Saqib, Shijing He, Junghyun Choy, Ruba Abu-Salma, Jose Such, Julia Bernd, and Mobin Javed. 2025. Bystander Privacy in Smart Homes: A Systematic Review of Concerns and Solutions. *ACM Trans. Comput.-Hum. Interact.* (May 2025). https://doi.org/10.1145/3731755 Just Accepted.

[60] Florian Schaub, Rebecca Balebako, Adam L. Durity, and Lorrie Faith Cranor. 2015. A Design Space for Effective Privacy Notices. In *Proceedings of the 2015 Symposium on Usable Privacy and Security (SOUPS '15)*. USENIX Association, Ottawa, Canada, 1–17.

[61] Timothée Schmude, Laura Koesten, Torsten Möller, and Sebastian Tschiatschek. 2023. On the Impact of Explanations on Understanding of Algorithmic Decision-Making. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency (FAccT '23)*. ACM, Chicago, IL, USA, 959–970. https://doi.org/10.1145/3593013.3594054

[62] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems* 35 (2022), 25278–25294.

[63] Andrew D. Selbst, Danah Boyd, Sorelle A. Friedler, Suresh Venkatasubramanian, and Janet Vertesi. 2019. Fairness and Abstraction in Sociotechnical Systems. In *Proceedings of the 2019 ACM Conference on Fairness, Accountability, and Transparency (FAT* '19)*. ACM, Atlanta, Georgia, USA, 59–68. https://doi.org/10.1145/3287560.3287598

[64] Andrew Senior, Sharath Pankanti, Arun Hampapur, Lisa Brown, Ying-Li Tian, Ahmet Ekin, Jonathan Connell, Chiao Fe Shu, and Max Lu. 2005. Enabling video privacy through computer vision. *IEEE Security & Privacy* 3, 3 (2005), 50–57.

[65] Shawn Shan, Emily Wenger, Jiayun Zhang, Huiying Li, Haitao Zheng, and Ben Y Zhao. 2020. Fawkes: Protecting privacy against unauthorized deep learning models. In *29th USENIX security symposium (USENIX Security 20)*. 1589–1604.

[66] Divya Shanmugam, Fernando Diaz, Samira Shabanian, Michèle Finck, and Asia Biega. 2022. Learning to limit data collection via scaling laws: A computational interpretation for the legal principle of data minimization. In *2022 ACM Conference on Fairness, Accountability, and Transparency*. 839–849.

[67] Tanusree Sharma, Lin Kyi, Yang Wang, and Asia J Biega. 2024. " I'm not convinced that they don't collect more than is necessary":{User-Controlled} Data Minimization Design in Search Engines. In *33rd USENIX Security Symposium (USENIX Security 24)*. 2797–2812.

[68] Tanusree Sharma, Abigale Stangl, Lotus Zhang, Yu-Yun Tseng, Inan Xu, Leah Findlater, Danna Gurari, and Yang Wang. 2023. Disability-first design and creation of a dataset showing private visual information collected with people who are blind. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–15.

[69] Tanusree Sharma, Yu-Yun Tseng, Lotus Zhang, Ayae Ide, Kelly Avery Mack, Leah Findlater, Danna Gurari, and Yang Wang. 2025. "Before, I Asked My Mom, Now I Ask ChatGPT": Visual Privacy Management with Generative AI for Blind and Low-Vision People. In *To appear in 27th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2025)*.

[70] Xinyue Shen, Yun Shen, Michael Backes, and Yang Zhang. 2025. When GPT Spills the Tea: Comprehensive Assessment of Knowledge File Leakage in GPTs. *arXiv preprint arXiv:2506.00197* (2025).

[71] Varad Srivastava. 2024. Lending an Ear: How LLMs Hear Your Banking Intentions. In *Proceedings of the 5th ACM International Conference on AI in Finance (ICAIF '24)*. Association for Computing Machinery, New York, NY, USA, 301–309. https://doi.org/10.1145/3677052.3698608

[72] Daniel Susser. 2022. Decision Time: Normative Dimensions of Algorithmic Speed. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT '22)*. 3533198.

[73] Omer Tene and Jules Polonetsky. 2011. Privacy in the age of big data: a time for big decisions. *Stan. L. Rev. Online* 64 (2011), 63.

[74] Parinya Thetbanthad, Benjaporn Sathanarugsawait, and Prasong Praneetpolgrang. 2025. Automated Redaction of Personally Identifiable Information on Drug Labels Using Optical Character Recognition and Large Language Models for Compliance with Thailand's Personal Data Protection Act. *Applied Sciences* 15 (04 2025), 4923. https://doi.org/10.3390/app15094923

[75] Parinya Thetbanthad, Benjaporn Sathanarugsawait, and Prasong Praneetpolgrang. 2025. Automated Redaction of Personally Identifiable Information on Drug Labels Using Optical Character Recognition and Large Language Models for Compliance with Thailand's Personal Data Protection Act. *Applied Sciences* 15, 9 (2025). https://doi.org/10.3390/app15094923

[76] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. 2016. Yfcc100m: The new data in multimedia research. *Commun. ACM* 59, 2 (2016), 64–73.

[77] Yu-Yun Tseng, Tanusree Sharma, Lotus Zhang, Abigale Stangl, Leah Findlater, Yang Wang, Danna Gurari Yu-Yun Tseng, and Danna Gurari. 2024. BIV-Priv-Seg: Locating Private Content in Images Taken by People With Visual Impairments. *arXiv preprint arXiv:2407.18243* (2024).

[78] Bryan Wang, Yuliang Li, Zhaoyang Lv, Haijun Xia, Yan Xu, and Raj Sodhi. 2024. Lave: Llm-powered agent assistance and language augmentation for video editing. In *Proceedings of the 29th International Conference on Intelligent User Interfaces*. 699–714.

[79] wikipidia. Peak signal-to-noise ratio. https://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio. (????). Accessed: 2025-09-10.

[80] Yuan-Jhe Yin, Bo-Yu Chen, and Berlin Chen. 2024. A Novel LLM-based Two-stage Summarization Approach for Long Dialogues. In *2024 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 1–6.

[81] Chiyuan Zhang, Daphne Ippolito, Katherine Lee, Matthew Jagielski, Florian Tramèr, and Nicholas Carlini. 2023. Counterfactual memorization in neural language models. *Advances in Neural Information Processing Systems* 36 (2023), 39321–39362.

[82] Zhiping Zhang, Michelle Jia, Bingsheng Yao, Sauvik Das, Ada Lerner, Dakuo Wang, and Tianshi Li. 2023. It'sa fair game, or is it? examining how users navigate disclosure risks and benefits when using llm-based conversational agents. *arXiv preprint arXiv:2309.11653* (2023).

[83] Jijie Zhou, Eryue Xu, Yaoyao Wu, and Tianshi Li. 2025. Rescriber: Smaller-LLM-Powered User-Led Data Minimization for LLM-Based Chatbots. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. 1–28.

[84] Shoshana Zuboff. 2019. *The age of surveillance capitalism: The fight for a human future at the new frontier of power*. Profile Books.

A Threat Model

Modern document workflows involve outsourcing tasks to third-party services such as PDF editors, cloud storage systems, and large language models (LLMs). Users upload documents for tasks like merging, splitting, editing, or summarization. While convenient, this creates a substantial privacy and security risk: sensitive data such as personal identifiers, financial information, or proprietary content can be inadvertently exposed to untrusted environments. Our threat model therefore considers how sensitive content may be exposed in each workflow stage, and how our proposed redact-and-restore mechanism reduces the attack surface.

In modeling threats to document workflows, it is important to first consider the categories of assets that require protection. At the core is the document content itself, including the raw text and embedded media such as names, identification numbers, contracts, or financial records. Closely tied to this is the often overlooked layer of metadata: filenames, timestamps, authorship information, and device identifiers can all leak contextual clues about the user or their organization. There is also derived data to consider. Many tools generate secondary artifacts such as vector embeddings, summaries, or structured analytics that may expose sensitive information. Finally, the most critical assets are the keys and access controls that govern restoration and authentication. If adversaries compromise these, they could bypass obfuscation and directly recover protected data.

Against these assets, several classes of adversaries must be anticipated. External attackers, including eavesdroppers and man-in-the-middle actors, pose a threat during file upload or transmission by attempting to intercept the document in transit. Service providers themselves, such as editing tools or LLM platforms, also represent a risk: through intentional logging, routine debugging, or retention policies, they may store or analyze sensitive content beyond its intended purpose. In addition, insider threats must be considered, malicious employees or contractors within service providers who have privileged access and can misuse it for data extraction. Finally, the rise of large language models introduces a novel adversary strategy which is adversarial prompts. By providing manipulative instructions, attackers can trick an LLM into revealing sensitive portions of a document. It opens up a new vector of data leakage unique to model-based workflows.

Table 7. Threats across editing tools and mitigations in RedactoRestore.

Step	Threat	Mitigation
Upload	Interception	Obfuscated upload only
Processing	Insider access, logs	Process obfuscated text only
Storage	Long-term retention	Vault separate; local key only
Output	Tampering	Local restoration map check

A.1 PDF Editing Tools

Users upload files into cloud-based PDF editors for merge, split, or annotation tasks. Threats arise at different stages: upload, processing, storage, and output delivery.

Upload. During upload, the primary threat is interception in transit. Adversaries positioned on the network (e.g., man-in-the-middle attackers) can capture the entire file, including identifiers, contracts, or medical information. Even when transport encryption is in place, risks remain from misconfigured TLS, certificate injection, or compromised endpoints. The impact is immediate disclosure of the document in raw form.

Processing. At the processing stage, the document is actively handled by a third-party service. Threats include unauthorized internal access (e.g., service provider employees viewing logs), and persistent storage in debugging or analytics pipelines. These threats can lead to long-term disclosure beyond the intended task.

Storage. Many services retain uploaded documents, either deliberately (for versioning, analytics) or inadvertently (in caches, backups, or logs). The threat here is long-term retention and forensic recovery. Even if a user deletes a file, remnants may survive in server snapshots, archival systems, or replicated databases. This poses a particularly high risk for regulated data (e.g., GDPR, HIPAA), as disclosure can persist long after initial processing.

Output Delivery. In the final stage, the user receives the processed output. Threats include tampering with the returned file (e.g., attackers injecting malicious macros or modifying the document) and subtle content injection (adding misleading or adversarial data to influence later use). The impact extends beyond disclosure, potentially compromising integrity and leading to downstream harm if the altered file is trusted.

A.2 LLM Services

Users also upload documents to large language model (LLM) services for tasks such as summarization, question answering, or semantic extraction. These workflows introduce a

distinct set of threats compared to PDF editing tools, particularly due to the nature of model training, embeddings, and prompt interactions.

Upload. At the upload stage, the entire text of the document is sent to the provider. The primary threat is that the provider gains access to raw, confidential data. Even if privacy policies claim otherwise, uploaded text may be logged, retained for fine-tuning, or exposed to insider access. The impact is direct disclosure of sensitive content to an untrusted service.

Context Injection. During interaction, adversaries may exploit the model with specially crafted prompts to reveal content from redacted regions. This class of threat, known as *prompt injection*, can bypass intended safeguards and expose hidden information. The impact is targeted leakage of sensitive fragments that were not supposed to be revealed.

Embedding Storage. Many LLM services store vector embeddings of uploaded text for retrieval, personalization, or analytics. The threat here is that sensitive information, even if partially masked, becomes encoded and retained in vector databases. Since embeddings can sometimes be inverted or deanonymized, the risk is long-term disclosure of private information.

Model Outputs. Finally, model outputs themselves can present risks. Even when the input text is obfuscated, models may attempt to reconstruct or infer hidden content through reasoning or pattern completion. Summaries may inadvertently reveal private information if placeholders are contextually obvious. This threatens both confidentiality (through unintended disclosure) and integrity (if outputs blend accurate and fabricated details).

Across workflows, the primary risk is confidentiality loss. Documents often contain personally identifiable information (PII), intellectual property, or regulated data. Integrity threats include the risk of tampering with processed files, while availability threats are less critical but possible.

The RedacToRestore mechanism provides two fundamental security benefits: *Minimized exposure.* Only obfuscated content is processed by third-party tools or LLMs, ensuring that raw sensitive information never leaves the user's trust boundary. *User-controlled restoration.* The mapping between obfuscated placeholders and original content is stored securely under user control, preventing unauthorized reconstruction by external services.

A.3 Prototype Implementation

The implementation of a functional prototype has followed the above design, featuring a graphical user interface to guide users through the redaction and restoration process. First, the user uploads a file in the interface. Upon file ingestion, the system automatically runs the Ingestion & Preprocessing module, determining whether the file is PDF or image-based. For PDFs, a pdf-reader extracts the text layer; if unavailable, an OCR engine (Tesseract) is invoked. A document layout model then segments the page structure and extracts coordinates for text elements, tables, and embedded figures. For images, a YOLOv10-based pipeline provides region proposals which are refined by image preprocessing before segmentation. Once preprocessing is complete, the user is prompted to scan for sensitive information. Under the hood, the prototype uses a fine-tuned BERT-based PII detection model to classify text entities such as names, emails, phone numbers, addresses, and identification numbers. For visual content, the system employs an image model trained on BIV-Priv-Seg to identify privacy-sensitive regions. For face detection, we integrated the OpenCV Haar Cascade Classifier [47], a lightweight but robust method for locating frontal faces. This choice allows the system to identify faces even in mixed-layout documents without adding significant inference overhead. For logo detection, we adopted the YOLOv8-based OpenLogo fine-tuned model [57] from Hugging Face. This detector is trained on a wide set of corporate and institutional logos, making it appropriate for spotting brand marks or organizational symbols that often carry sensitive associations. Both detectors output bounding boxes that are merged with text and segmentation results. All candidate redactions are unified through a coordinate mapping step, producing a set of redactable elements that are then passed to the subsequent modules.

After scanning, the prototype presents the results in an accessible UI. Detected sensitive content is highlighted in the preview and summarized by category in a sidebar (Figure 3). For example, if the file contains an email and a phone number, the sidebar will list "Email Addresses (1)" and "Phone Numbers (1)" with the found instances listed under each. At this stage, users can choose how to proceed with redaction. A combination of *Select All* and *Redact File* button allows one-click removal of all detected items using default settings (e.g., black bar overlay for text and removal of all metadata). Alternatively, the user can click *Customize* to adjust the redaction settings. In the customization view (Figure 3), each sensitive item can be individually selected or deselected for redaction, giving users granular control over what to redact. The user can also select the redaction style: our prototype supports placing black bars over text, replacing text with blurred boxes, or substituting the text with a *[REDACTED]* label. Each style provides a different visual cue when the original text is actually hidden. This supports a range of privacy preferences, from maximum redaction to selective redaction.

Once the user confirms their choices and clicks *Redact File*, the system performs redaction and securely stores the data. Implementation-wise, the prototype generates a redacted version of the file in memory. For each selected sensitive text item, the system masks the text and overlays the chosen obfuscation. For instance, if the user chose black bar redactions, it draws opaque black rectangles over the exact coordinates

Table 8. Threats and mitigations across LLM workflows.

Step	Type	Description
Upload	Threat	Provider gains raw, confidential data.
	Mitigation	Only obfuscated version is uploaded.
Context Injection	Threat	Prompt injection leaks hidden data.
	Mitigation	Only placeholders are processed; sensitive text hidden.
Embedding Storage	Threat	Sensitive info retained in vector DB.
	Mitigation	Embeddings are computed from obfuscated text, not raw PII.
Model Outputs	Threat	Summaries may reconstruct private info.
	Mitigation	Restoration occurs only client-side with user key.

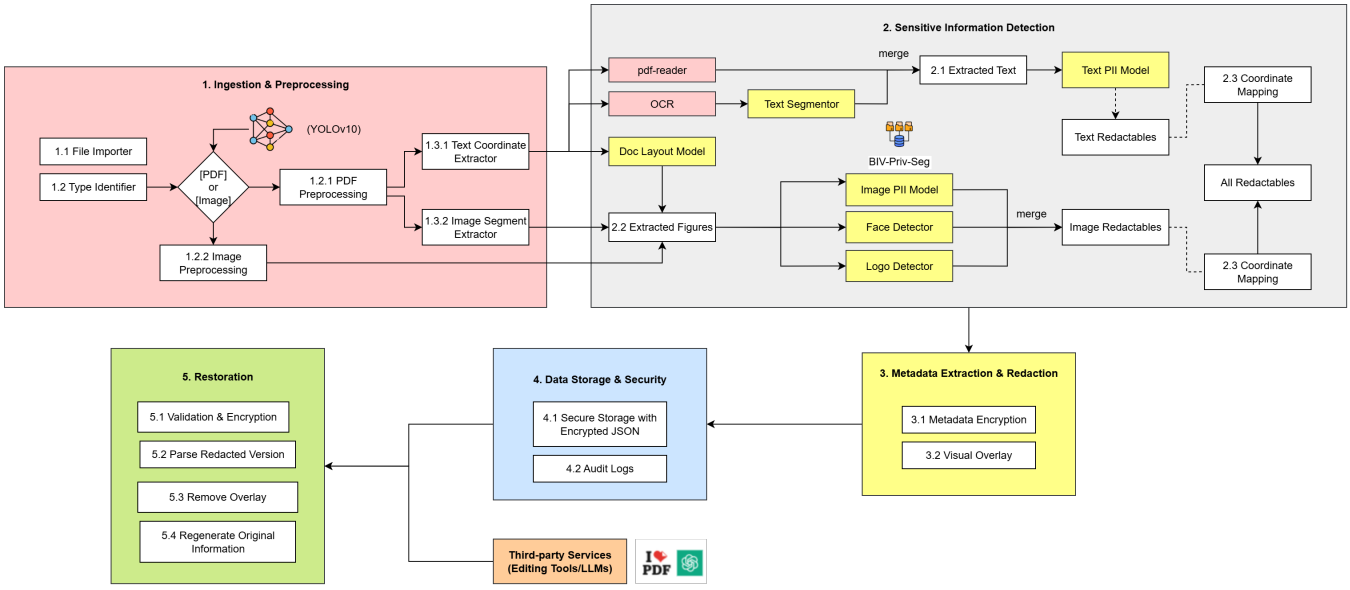

Figure 7. Overview of the **RedactoRestore** system architecture. The workflow is divided into five modules: (1) *Ingestion & Preprocessing*, which imports the file, classifies its type, and extracts text or image regions; (2) *Sensitive Information Detection*, combining text PII models, image segmentation, face and logo detectors with a coordinate mapping stage; (3) *Metadata Extraction & Redaction*, encrypting metadata and applying visual overlays to redactables; (4) *Data Storage & Security*, storing original content in encrypted JSON with audit logs; and (5) *Restoration*, which validates, removes overlays, and reconstructs the original content after third-party processing.

of each sensitive word/phrase. If the *[REDACTED]* text option is chosen, the system replaces the original text string with the placeholder word formatted to match the original font size and style. In both cases, the original text is no longer present in the visible text layer. All original sensitive data is simultaneously recorded for storage: the prototype compiles a JSON object containing each redacted item's original text and its position, and its category/type. Before saving, the JSON is encrypted using a symmetric key algorithm where the key is derived from a system-generated secret kept locally. This encrypted JSON serves as a secure vault of the sensitive information. It logs the redaction actions including a timestamp and the types of data removed to an internal audit log for later reference. Finally, the redacted file is the output to the user. The modified file is automatically downloaded with a new name *(e.g., FileName_Redacted.pdf)*. A confirmation message notifies the user that redaction is complete. At this point, the user can forward the redacted version to the Third-party Services (TPS) for processing.

After the TPS have processed the redacted file, the user can invoke the restoration process to recover the original content. In this prototype, it is achieved by loading the processed version back into the tool and selecting a *Restore File* function. To ensure secure and reversible redaction, our prototype embeds information about the redacted regions in the PDF's metadata rather than directly overlaying it on the page. This design avoids a key limitation of conventional approaches: when a PDF is flattened (i.e., all layers merged into a single uneditable image), the entire page becomes unselectable. If

we do encoding on PDF pages, such actions will prevent the preservation of structural cues. In our system, sensitive text or images are removed from the visual layer so they cannot be scraped by third-party services with OCR or text parsers, but metadata containing only bounding-box coordinates can be retained. These coordinates are harmless on their own, as they do not include the original sensitive content, but they provide sufficient reference points to support future restoration by an authorized user. This approach balances privacy protection since no recoverable sensitive data remains in the flattened PDF with reversibility since redacted regions can later be reconstructed using the coordinate metadata. The Restoration module then uses the previously stored sensitive-data JSON to reconstruct the data. First, the prototype locates the encrypted JSON file corresponding to the file (using a file ID) and decrypts it after validating the user's keys if required. The system then parses it to identify the redaction markers. For black bar overlays, the prototype knows the coordinates from the JSON) and simply removes those drawn rectangle objects. For *[REDACTED]* text placeholders, it searches the file text for the placeholder strings. The mapped coordinates may not match as the processed file can have changes in pages or positions. So, it finds exact mapping through embedded invisible redaction ID. Once located, it re-inserts the original content: for each redacted item, the original text from the JSON is placed back into the exact position and with the original font styling. In case the TPS have added new content or comments to the PDF, our restoration procedure is designed to merge those changes with the original text. For example, by ensuring that newly added annotations remain while the underlying text is restored. The prototype also restores the file metadata by writing back any saved metadata fields. After removing all obfuscations and reinserting content, the output is a file identical to the file type as it existed before redaction. The sensitive JSON and any keys can then be disposed of, or retained for record-keeping, depending on user preference.