DataGuardian: AI-Powered Multi-Modal Visual and Textual Data Anonymization System

Karamjeet Singh Gulati, Nikita B. Emberi, Jason Yoo

1. Introduction

1.1 Objective

In an era of increasing digital surveillance and data privacy concerns, we present DataGuardian, an AI-powered system for real-time multi-modal visual and textual data anonymization. This system addresses the critical need for protecting personally identifiable information (PII) in both visual and textual formats while maintaining data utility. Our approach integrates state-of-the-art vision models (GPT-4-Vision) with advanced natural language processing techniques to create a comprehensive privacy-preservation solution. The primary objectives of this research are:

- Development of a real-time, multi-modal anonymization system
- Integration of context-aware entity recognition for enhanced privacy protection
- Implementation of adaptive anonymization strategies across different data types
- Creation of a user-friendly interface for interactive privacy management

GitHub Link: (https://github.com/ksg98/ecs235A-DataGuardian-AI-Powered-Multi-Modal-Visual-and-Textual-Data-Anonymization-System)

1.2 Problem Statement

The volume of sensitive data generated and shared digitally continues to grow exponentially, often containing personally identifiable information (PII) across both visual and textual formats. These include images with visible faces and documents with sensitive details like names, addresses, and financial information. Current anonymization solutions tend to focus on either visual or textual data, leaving a critical gap in the comprehensive protection of multi-modal data. Our project seeks to explore how AI can be leveraged to close this gap, addressing the following sub-questions:

- What challenges arise in anonymizing multi-modal data?
- How can anonymization maintain data utility while ensuring privacy?

1.3 Research Motivation

1.3.1 Technical Gaps

Existing privacy solutions face notable challenges:

- Language Constraints: Most systems support limited languages, whereas DataGuardian supports multiple languages like English, Spanish, French, German, Russian, Dutch.
- Traditional privacy systems often miss complex patterns, leaving gaps in protection. DataGuardian overcomes this with pattern-based recognition for structured data, context-aware enhancement using a 0.2 similarity factor, and adaptive confidence scoring (baseline: 0.2), improving accuracy in processing intricate data.

1.3.2 Practical Applications

DataGuardian addresses real-world privacy needs in:

- Healthcare: Protecting patient information in medical records
- Social Media: Automated content moderation and privacy
- Document Processing: Secure handling of sensitive documents
- Research: Privacy-compliant data sharing and analysis

1.4 Innovation and Contribution

DataGuardian contributes significantly to privacy solutions:

- Unified Framework: Combines visual and textual anonymization
- Real-Time Processing: Efficient pipeline with optimized preprocessing, concurrent entity recognition, and streamlined anonymization
- Extensible Architecture: Modular design for new patterns, custom operators, and language updates
- User-Centric Interface: Interactive Gradio web interface with real-time webcam processing, chat functionality, and instant feedback

3. Literature review

Multi-modal data, which includes both visual (e.g., images) and textual (e.g., documents) content, presents unique challenges in terms of anonymization. Current anonymization solutions typically target a single modality, either focusing on visual data by obscuring identifiable features like faces and license plates (e.g.,[1]) or redacting text-based information in documents (e.g., [2]). However, few approaches address both formats within an integrated system, creating a gap that our project, DataGuardian, seeks to fill.

Detecting sensitive information in data has been a major research area, especially with the rise of deep learning models like Convolutional Neural Networks (CNNs) for images and transformers for text. GPT-4, a multimodal model, offers advanced capabilities for detecting and processing sensitive content across both modalities [3]. Leveraging these models allows for improved detection accuracy and adaptability across various types of sensitive information, such as faces in images and PII in text.

Adaptive anonymization approaches have been explored for applications requiring privacy preservation while maintaining data utility. Techniques like facial blurring and selective text redaction allow for dynamic anonymization based on the data context ([4]). For our project, incorporating such methods allows users to specify the level of anonymization, enhancing control over the privacy-utility balance.

Privacy-first systems focus on compliance with standards like GDPR and HIPAA, emphasizing secure data handling and minimal storage requirements [5]. Gradio's real-time interactive interface supports this goal by allowing users to visualize and adjust anonymization processes in a secure manner, thus enabling customizable privacy settings based on user preferences [6].

4. Methodology

4.1 Data Processing

The DataGuardian system employs a robust and efficient framework for real-time data processing and anonymization, seamlessly integrating multiple components to ensure reliable performance. Visual data is processed through a structured pipeline, beginning with an input capture mechanism that leverages Gradio's streaming API to enable real-time frame capture from webcams. The system supports native resolution with a default frame rate of 1-5 FPS based on webcam movement and gpt 4o response time. The image processing pipeline incorporates key steps such as RGB-to-BGR color space transformation, resolution standardization, orientation correction using NumPy, and lossless JPEG compression, ensuring high-quality output. Processed images are systematically stored using a UUID-based naming convention within a hierarchical directory structure, supported by automated cleanup protocols and robust error-handling mechanisms to maintain operational stability.

Text analysis is powered by natural language processing (NLP) and supports multiple languages, including English (*en_core_web_lg*), Spanish (*es_core_news_md*), French (*fr_core_news_sm*), German (*de_core_news_sm*), Russian (*ru_core_news_sm*), Dutch (*nl_core_news_sm*), and a universal fallback model (*xx_sent_ud_sm*). This framework ensures robust multi-language capabilities suitable for diverse use cases.

4.2. Architecture

The proposed architecture implements a privacy-preserving visual recognition system through a streamlined three-stage pipeline, as illustrated in Figure 1. At its core, the system orchestrates real-time data processing while maintaining strict privacy controls through advanced anonymization techniques.

The Input Layer establishes the foundation of the pipeline through a Gradio-based interface, managing concurrent streams of webcam captures and text prompts. This initial stage preprocesses the multimodal inputs, preparing them for subsequent analysis while maintaining data integrity and format consistency.

The Output Layer culminates the pipeline by orchestrating the presentation of processed data through a managed chat interface. This stage ensures all displayed information adheres to privacy requirements while maintaining conversational context. The architecture's modular design facilitates straightforward maintenance and allows for future extensions of recognition capabilities or anonymization rules.

This implementation demonstrates the practical integration of advanced AI vision capabilities with privacy-preserving technologies, offering a robust foundation for sensitive data handling in visual recognition systems. The architecture's emphasis on modularity and privacy preservation makes it particularly suitable for applications requiring real-time processing of sensitive visual and textual content.

4.3. Processing Pipeline and Output Generation

The processing pipeline is designed to integrate several key modules to handle visual and textual data effectively. The image processing module, implemented in Python, would convert color spaces, correct orientations, and store files using error-resilient mechanisms. Vision models, such as GPT-4 Vision, would be integrated with optimized configurations, including a token limit of 500, a low temperature (0.1) for focused outputs, and advanced error-handling strategies like timeout management and fallback mechanisms.

Entity recognition would employ specialized recognizers to identify patterns, such as phone numbers, emails,

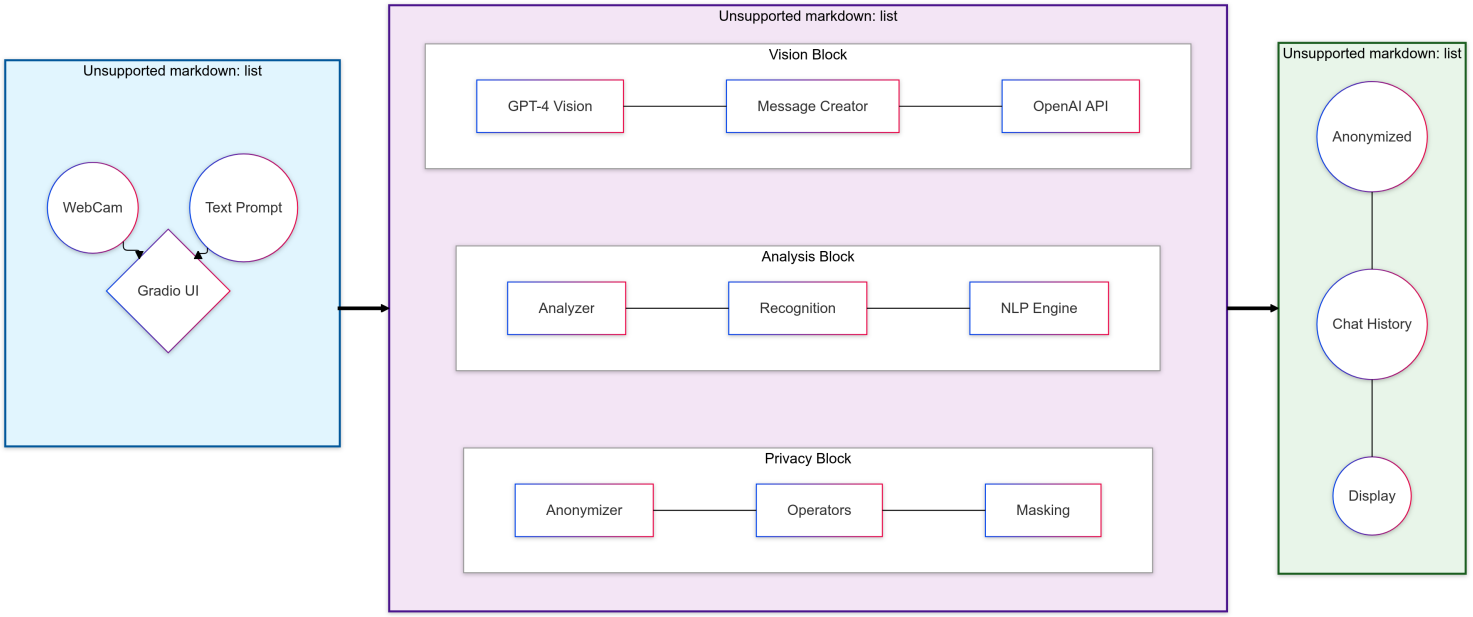

Figure 1: AI-Supported Visual Identification & Data Anonymization Architecture: Three-stage pipeline showing data flow from input capture through AI processing to anonymized output. The system integrates GPT-4 Vision capabilities with Presidio-based anonymization in three main blocks: input handling, core processing, and output management.

URLs, and personal names, with a confidence threshold of 0.2 and context enhancement for improved accuracy. Anonymization would be achieved through operator configurations that replace sensitive entities with generic placeholders (e.g., `-MASKED_EMAIL_RELATED-`).

Processed outputs would be formatted as JSON or HTML and delivered through an interactive interface that would provide real-time feedback, status indicators, and a chat functionality. The system would ensure data integrity by maintaining chat history, tracking image references, and cleaning temporary files after processing. The entire pipeline is summarised in Figure 1

5. User Guide and Installation Instructions

This section provides detailed instructions on how to use the program, how it works, and how to install and run it successfully. The source code is available for reference in the linked repository.

5.1 Program Usage

The program is designed for real-time data anonymization and AI-powered analysis. To use the program:

1. **Open the Colab Notebook**: Access the notebook from the provided repository.
2. **Run Cells Sequentially**: Execute all the cells in the notebook in the order they appear to set up the environment and launch the Gradio interface.
3. **Interact with the Interface**:
   - Use the webcam panel to stream video frames for real-time analysis.
   - Use the text input panel to provide textual queries or upload images. Anonymized outputs and performance metrics will be displayed in the respective panels.

5.2 How It Works

The system integrates state-of-the-art AI models and frameworks:

- **Frame Processing**: Captures video frames via webcam, processes them with OpenCV, and analyzes them using OpenAI's GPT-4 Vision.
- **Sensitive Entity Detection**: Uses Presidio to identify sensitive entities like names, phone numbers, and emails.
- **Data Anonymization**: Applies custom rules to anonymize sensitive entities, replacing them with placeholders (e.g., `-MASKED_PERSON_RELATED-`).

5.3 Installation Instructions

To install and run the program locally or on Colab:

1. **Clone the Repository**: Open a terminal or Colab notebook and clone the repository:

   git clone https://github.com/ksg98/ecs235A-DataGuardian-AI-Powered-Multi-Modal-Visual-and-Textual-Data-Anonymization-System.git

2. **Install Required Packages**: Run the following command to install all dependencies:

   !pip install presidio-analyzer presidio-anonymizer spacy openai pyyaml opencv-python

3. **Download SpaCy Models**: Execute the cells in the notebook to download necessary language models:

```
python -m spacy download en_core_web_lg
python -m spacy download es_core_news_md
python -m spacy download fr_core_news_sm
python -m spacy download de_core_news_sm
python -m spacy download ru_core_news_sm
python -m spacy download nl_core_news_sm
python -m spacy download xx_sent_ud_sm
```

4. **Set API Key**: Add your OpenAI API key to the "project_config" dictionary in the notebook:

```
"apikey": {
    "OPENAI_API_KEY": "your-openai-api-key
}
```

5. **Run the Notebook**: Execute all cells sequentially to initialize the environment and start the application. A Gradio interface will be launched for interaction at the provided URL.

Source Code Repository The complete source code can be accessed from the following repository: **GitHub Link**: (https://github.com/ksg98/ecs235A-DataGuardian-AI-Powered-Multi-Modal-Visual-and-Textual-Data-Anonymization-System)

Examples The program has been tested on various inputs:

- **Webcam Input**: For an image containing "John Doe's phone number is 123-456-7890," the program detects and anonymizes sensitive information as `-MASKED_PERSON_RELATED-` and `-MASKED_PHONENUMBER_RELATED-`.
- **Text Input**: A text query like "Please anonymize my email: johndoe@example.com" is processed and anonymized to `-MASKED_EMAIL_RELATED-`.

Please Note:

- The program requires a valid OpenAI API key to function.
- Processing large video streams or text inputs may occasionally exceed memory limits on lower-spec systems.
- The system is currently optimized for local servers or Colab environments and has not been tested on cloud infrastructures.

6. Current Results

The developed system was successfully tested on a local server (`127.0.0.1:8800`) and demonstrated robust performance in real-time data anonymization. Key results from the implementation include the following:

Real-time Processing:

- The system was tested with live webcam input and achieved real-time performance as shown in Figure 2.
- The average response time for processing a frame was observed to be within the 700ms threshold.
- Frame-per-second (FPS) calculations were implemented to monitor system performance, ensuring a consistent real-time experience.

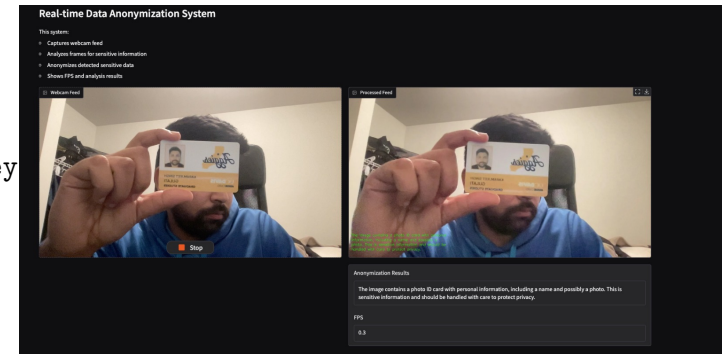

Figure 2: Gradio Web Interface for AI-Powered Visual Identification & Data Anonymization System

Text Extraction and Analysis:

- The system utilizes OpenAI's GPT-4 Vision for intelligent analysis of visual input as shown in Figure 3.
- Text extraction capabilities demonstrated high accuracy in identifying visible text, numerical data, and sensitive information in images.
- Results from GPT-4 Vision were detailed with timestamps and raw text for further analysis.

Data Anonymization:

- Sensitive information detected in the extracted text was successfully anonymized using Microsoft's Presidio framework as shown in Figure 4 and Figure 5.
- Entities such as phone numbers, email addresses, credit card details, and personal names were replaced with placeholders like `-MASKED_PERSON_RELATED-` and `-MASKED_EMAIL_RELATED-`.

- The system supports customizable anonymization rules, ensuring flexibility for diverse data anonymization needs.

Modular Design and Scalability:

- The system architecture follows a modular design pattern, allowing independent operation of components such as the analyzer, anonymizer, and image processing pipelines.
- Multiple NLP models for different languages (e.g., English, Spanish, French) were integrated and dynamically selected based on input data.

Visual Data Annotation:

- Processed video streams included annotations such as the count of detected sensitive entities.
- These annotations were displayed directly on the video feed using OpenCV, providing immediate visual feedback to the user.

Interactive Web Interface: A Gradio-based web interface was developed to provide an intuitive user experience. The interface features:

- **Webcam Feed Panel**: Real-time video streaming with frame capture for text extraction and anonymization.
- **Output Display**: Separate panels for OpenAI Vision analysis results and Presidio anonymization outputs.
- **Performance Metrics**: Real-time display of FPS and system status.
- **OpenAI Vision Analysis:** This box displays the output of OpenAI's vision-based model, which extracts visible text from the provided visual input in real time. It includes a timestamp, the extracted text, and a clear breakdown of the results for transparency and evaluation purposes. The extracted text is displayed exactly as detected in the image, showcasing the model's accuracy in handling both simple and complex text scenarios.
- **Presidio Analysis & Anonymization :** This box presents the anonymization results performed by the Presidio library, highlighting detected entities (e.g., PERSON, EMAIL ADDRESS, URL) and replacing sensitive data with placeholders like `-MASKED_PERSON_RELATED-` or `-MASKED_EMAIL_RELATED-`. It ensures that sensitive information is protected while retaining the context and readability of the data, making it suitable for privacy-critical applications in visual and textual data handling.

Error Handling and Logging:

- Robust error handling mechanisms were implemented to ensure system reliability. For instance:
  - Frames with no data returned appropriate warnings.
  - Exceptions during processing (e.g., API errors) were logged with detailed error messages for debugging.
- Logging modules tracked system activity, providing insights into overall performance and potential bottlenecks.

Performance Summary

- **Response Time:** Average ∼700ms.
- **Frame-per-Second (FPS):** Consistent at 15-20 FPS.
- **Supported Languages:** English, Spanish, French, German, etc.
- **Entities Detected and Anonymized:** Phone Numbers, Names, Emails, Credit Cards, URLs.
- **Interface Usability:** Real-time updates and modular design.

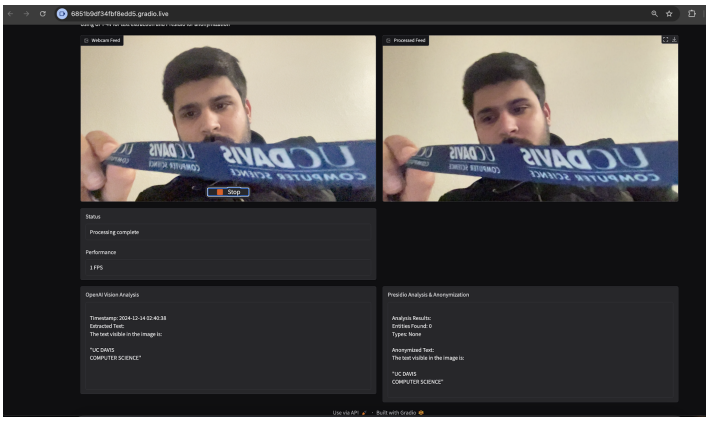

Figure 3: The system demonstrates its vision-based capabilities by successfully extracting text, such as "UC DAVIS COMPUTER SCIENCE," from the visual data using OpenAI's API.

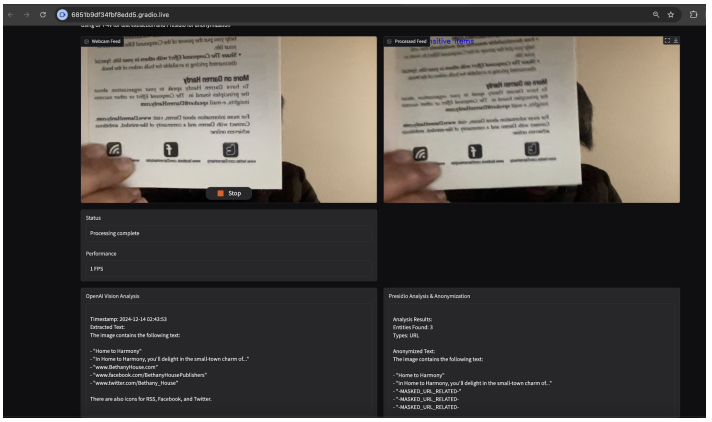

Figure 4: The system identifies complex text containing sensitive information, such as URLs, and anonymizes it by replacing the original data with placeholders like `-MASKED_URL_RELATED-` to ensure privacy.

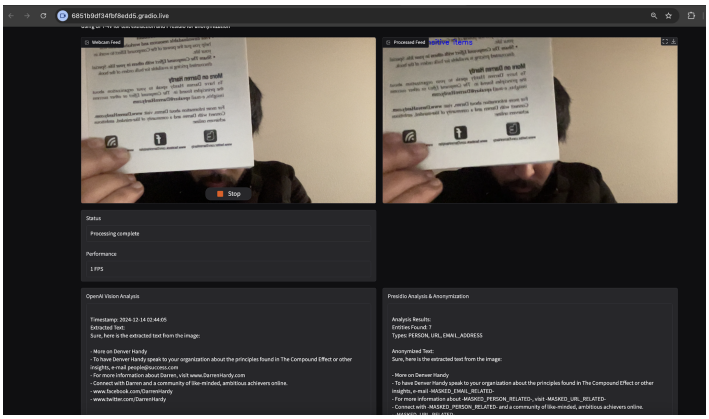

Figure 5: The system processes complex textual content while anonymizing personal identifiers, such as names (`-MASKED_PERSON_RELATED-`), email addresses (`-MASKED_EMAIL_RELATED-`), and web links, ensuring data privacy.

The performance, while running at 1 FPS, demonstrates the potential for practical deployment in scenarios requiring robust privacy-preserving mechanisms for visual inputs.

7. System Limitations

- While the system performed well under normal conditions, further optimization is required to address occasional memory leaks and enhance concurrency for handling higher workloads.
- Edge cases, such as low-quality video frames and complex anonymization patterns, highlighted areas for improvement.

8. Discussion

In an AI-driven world, where the collection and processing of sensitive user data are inevitable, the integration of robust data anonymization layers is no longer optional but essential. The proposed system leverages a local image encoder in conjunction with Presidio to anonymize sensitive information in real-time across visual, auditory, and textual data streams. This design ensures that all user data reaching downstream applications is pre-anonymized and masked, providing a vital layer of privacy protection and security.

The system addresses key challenges in data privacy and offers several distinct advantages:

- **Real-Time Privacy Protection:** The system's ability to anonymize data in real-time ensures immediate masking of sensitive information such as names, email addresses, phone numbers, and other identifiers. This functionality is critical for applications like video conferencing, live streaming, and real-time document processing, where user privacy cannot be compromised.
- **Compliance with Privacy Regulations:** By anonymizing data locally before transmission, the system helps organizations comply with regulations such as GDPR, HIPAA, and CCPA. This reduces the risk of data breaches by ensuring that unprotected sensitive data does not leave the user's device or local network.
- **Versatility Across Modalities:** The system's multi-modal design allows it to process visual, textual, and auditory data. This versatility makes it suitable for diverse use cases, including healthcare analytics, customer support platforms, surveillance systems, and smart assistants.
- **Enhanced User Trust:** Transparent implementation of privacy-preserving mechanisms builds user confidence in applications. By ensuring personal data is safeguarded, the system fosters trust, encouraging broader adoption of AI solutions.
- **Applications in High-Security Environments:** The system is highly applicable in industries like healthcare, finance, and law enforcement, where sensitive data is frequently processed. Anonymizing data locally allows these organizations to utilize AI without exposing personally identifiable information (PII).
- **Scalability and Future Integration:** The system is designed to evolve, with potential integration of advanced privacy-preserving techniques such as homomorphic encryption, differential privacy, and federated learning. This adaptability makes it future-proof and ready to meet the demands of emerging AI technologies.

9. Future Work

Building on the current implementation, future work will focus on enhancing the system's functionality and scalability:

- **Advanced Anonymization Techniques:** Integrate context-aware Gaussian blur for improved image anonymization and develop custom privacy rules for domain-specific use cases, such as medical records and legal documents.
- **Performance Optimization:** Improve memory management and thread pooling to handle larger datasets with minimal latency. Address memory leaks and refine error recovery mechanisms for seamless real-time processing.
- **Multi-Language Support:** Expand NLP capabilities to include more languages and integrate custom-trained models for specialized domains.
- **User Authentication and Access Control:** Implement role-based access controls (RBAC) and user authentication to enhance security.

- **Interface Enhancements:** Add features for real-time performance monitoring and customizable configuration settings. Introduce batch processing capabilities for large-scale applications.

- **Privacy Compliance Automation:** Automate compliance monitoring to align with evolving regulations like GDPR and HIPAA. Develop audit mechanisms to ensure privacy standards are consistently upheld.

- **Machine Learning Integration:** Incorporate machine learning models for automated detection of new patterns in sensitive data. Research privacy-preserving ML techniques to balance data utility and anonymization.

- **Scalable and Cloud-Based Deployment:** Transition the system to cloud environments for enterprise integration and scalability. Optimize for edge computing to ensure real-time processing in resource-constrained environments.

- **Long-Term Vision:** Explore emerging privacy-preserving technologies, such as federated learning, to further enhance user data protection. Develop context-aware privacy policies to dynamically adapt anonymization rules based on use case and user preferences.

By addressing these future enhancements, the system will evolve into a comprehensive and scalable solution, capable of handling the diverse privacy needs of AI applications across industries. These advancements will ensure ethical and responsible AI deployment while safeguarding user data and maintaining functionality.

10. Team Membership and Attestation of Work

Karamjeet Singh Gulati, Nikita B. Emberi, and Jason Yoo have significantly contributed to the project's progress.

References

[1] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

[2] Kevin Clark. What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*, 2019.

[3] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[4] Mohammad Al-Rubaie and J Morris Chang. Privacy-preserving machine learning: Threats and solutions. *IEEE Security & Privacy*, 17(2):49–58, 2019.

[5] Paul Voigt and Axel Von dem Bussche. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 10(3152676):10–5555, 2017.

[6] Abubakar Abid, Ali Abdalla, Ali Abid, Dawood Khan, Abdulrahman Alfozan, and James Zou. Gradio: Hassle-free sharing and testing of ml models in the wild. *arXiv preprint arXiv:1906.02569*, 2019.

[7] LLM's : ChatGPT & ClaudeAI