

SafeRecords, una solución de software para la anonimización de registros médicos usando modelos de reconocimiento de entidades

Daniel David Gutierrez Cantillo, Santiago Andres Mercado Barandica, David Daniel Henriquez Leal
ddgutierrez@uninorte.edu.co, sabarandica@uninorte.edu.co, ddhenriquez@uninorte.edu.co

I. INTRODUCCIÓN

Hoy en día, proteger la información personal en los registros médicos es un tema de vital importancia en el campo de la salud y la investigación. Esto no solo protege la privacidad de los pacientes, sino que también permite usar los datos de forma segura en proyectos de investigación (Juez-Hernández et al., 2023). Es crucial desarrollar métodos efectivos para la anonimización de las notas clínicas. Si esto no se hace correctamente, puede acarrear graves consecuencias como la filtración de datos personales o el incumplimiento de leyes como el GDPR (Chevrier et al., 2019).

La inteligencia artificial ha sido muy útil para proteger la privacidad en los textos médicos en estudios recientes, en especial los modelos basados en BERT. Mao y Liu (2019) crearon un modelo llamado BERT-CRF que ha dado excelentes resultados, mientras que Wang y E (2021) utilizaron BERT para proteger la privacidad en registros médicos en chino, anonimizando la información personal. Además, Juez-Hernández et al (2023) resaltan que utilizar inteligencia artificial para anonimizar registros clínicos de forma automática no solo es más rápido, sino también más preciso, lo cual es una prioridad cuando hay un gran volumen de datos.

Teniendo en cuenta esta necesidad e inspirados por estos estudios, este proyecto busca desarrollar una solución de software que permita proteger la privacidad en los registros médicos. Se va a desplegar un modelo BERT directamente en el navegador web para identificar y eliminar la información personal por medio del reconocimiento de entidades. Este enfoque no solo garantiza la privacidad al evitar la transferencia de datos a servidores externos, sino que también ofrece una solución accesible y fácil de usar para profesionales de la salud. El objetivo es implementar una solución que sea precisa, fácil de usar y segura, para ayudar a proteger los datos médicos utilizando tecnología moderna de inteligencia artificial.

II. DESCRIPCIÓN DEL PROBLEMA

Con el número creciente de datos y registros digitales en el sector de la salud, han aumentado los retos para mantener la privacidad de los pacientes. Es de vital importancia anonimizar estos registros médicos para que puedan ser utilizados en investigaciones científicas sin comprometer la confidencialidad. Sin embargo, durante el proceso de eliminar información personal se puede afectar la utilidad y confiabilidad de los datos para uso en investigación (Carmona, Conesa, & Casas-Roma, 2019). En los últimos años, se ha conseguido mejorar el proceso de eliminación de datos personales en notas clínicas por medio de modelos de reconocimiento de entidades, como BERT, en registros médicos (Wang & E, 2021). A pesar de esto, la complejidad inherente al lenguaje y la gran variedad de datos personales en registros clínicos siguen suponiendo desafíos a superar (Pépin & Zulkernine, 2022).

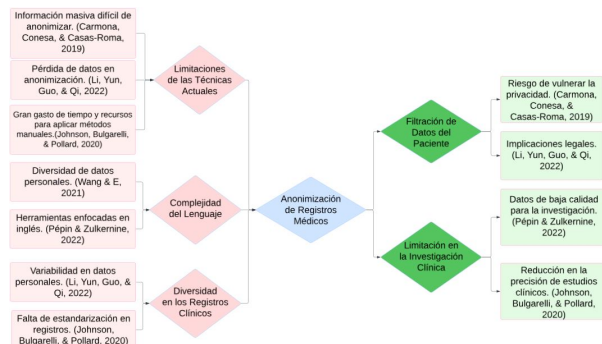


Figura 1: Árbol del problema

En la figura del árbol del problema se ilustran las limitaciones de las técnicas actuales de anonimización, y las consecuencias derivadas de estas. La complejidad del lenguaje y la diversidad tanto de registros clínicos como de datos personales conducen a problemas importantes como la disminución de la utilidad para investigaciones y la filtración de datos personales. De esta manera, se hacen evidentes los desafíos inherentes

a la implementación de una solución que sea segura y efectiva. Hoy en día existen muchas técnicas para proteger la privacidad, aunque muchas de ellas no son tan eficientes y precisas. Por lo tanto, es necesario un método que mantenga un equilibrio entre proteger la privacidad y preservar la utilidad de los datos para lograr el resultado deseado.

Los avances recientes en inteligencia artificial, particularmente en el procesamiento del lenguaje natural, han impulsado el desarrollo de nuevas herramientas para la anonimización de datos médicos, enfrentando los desafíos que presentan la privacidad y la utilidad de los datos. La anonimización es crucial para garantizar la protección de los pacientes y cumplir con regulaciones como HIPAA, mientras se preserva la integridad de los datos para su uso en investigación (Li, Yun, Guo, & Qi, 2022). Sin embargo, existen limitaciones en las técnicas actuales, como la pérdida de información relevante o el gasto significativo de tiempo y recursos, especialmente en la implementación de métodos manuales (Carmona, Conesa, & Casas-Roma, 2019). El lenguaje especializado en los registros médicos, además de la diversidad de formatos y datos personales, plantea dificultades adicionales para garantizar una anonimización efectiva (Wang & E, 2021). Herramientas desarrolladas con modelos avanzados como BERT, que permiten un procesamiento más contextualizado del lenguaje, se presentan como soluciones viables para automatizar la anonimización en un entorno multilingüe, superando las limitaciones de herramientas tradicionales que suelen centrarse en el inglés (Pépin & Zulkernine, 2022). Este proyecto busca implementar estas innovaciones para crear una solución eficiente, asegurando tanto la privacidad del paciente como la utilidad de los datos anonimizados en investigaciones clínicas y análisis posteriores.

III. JUSTIFICACIÓN

La anonimización de registros médicos no solo es crucial desde una perspectiva técnica, sino que también es fundamental para cumplir con las normativas de privacidad y fomentar el avance de la investigación científica. Como señalan Carmona, Conesa y Casas-Roma (2019), la falta de protección adecuada en los registros médicos podría comprometer tanto la confianza del paciente como la integridad del sistema de salud. En un entorno cada vez más digitalizado, el acceso a registros médicos anonimizados es vital para que investigadores puedan analizar grandes volúmenes de datos clínicos sin infringir las leyes de privacidad, como el Reglamento General de Protección de Datos (GDPR) en Europa o la Ley de Portabilidad y Responsabilidad de

Seguros de Salud (HIPAA) en los Estados Unidos. Sin embargo, garantizar la privacidad mientras se conserva la utilidad de los datos para investigaciones científicas sigue siendo un desafío apremiante (Kovacevic et al, 2023). En este sentido, las técnicas de reconocimiento de entidades nombradas (NER), apoyadas por modelos avanzados como BERT, han demostrado ser esenciales para eliminar automáticamente información personal sin comprometer la riqueza informativa de los datos. (Johnson et al., 2020; Li et al., 2022).

Siguiendo esta idea, el proyecto encuentra su razón de ser en la necesidad de una solución automatizada para proteger la privacidad al eliminar datos personales, al mismo tiempo que se mantiene utilidad de los datos para la investigación médica. Teniendo esto en cuenta, este proyecto ofrece una herramienta eficiente y accesible para la anonimización de registros médicos, dirigida a instituciones de salud y a investigadores que buscan proteger la privacidad de los pacientes sin comprometer la integridad de los datos. Se hará uso de BERT para el reconocimiento de entidades, ya que permite identificar y eliminar de manera automática la información sensible con un alto grado de precisión (Xu et al., 2021).

IV. OBJETIVOS

IV-A. *Objetivo general*

Modelar, diseñar e implementar una solución de software para la anonimización de registros médicos, que reemplace los datos personales por medio de procesamiento de reconocimiento de entidades usando BERT.

IV-B. *Objetivos específicos*

- Elaborar la revisión sistemática de la literatura, abarcando temas relacionados con el objeto de estudio, como lo son el despliegue web de modelos de inteligencia artificial, modelos de reconocimiento de entidades, BERT, anonimización de textos, entre otros.
- Desarrollar la arquitectura de la solución asociada al software para anonimizar registros médicos.
- Implementar el prototipo de solución asociado al software para anonimizar registros médicos.
- Validar el prototipo de la solución de software y la estructura asociada al mismo.

V. METODOLOGÍA DE INVESTIGACIÓN

La metodología de investigación del proyecto de anonimización de registros médicos mediante reconocimiento de entidades con BERT se estructura en cuatro fases. En primer lugar, se realiza una revisión de la literatura para identificar estudios

relevantes sobre la privacidad de datos y técnicas de anonimización de información. Luego, se diseña un modelo IT enfocado en la anonimización eficiente de los registros médicos. En la siguiente fase, se desarrolla un prototipo funcional de la solución, que finalmente es validado en términos de usabilidad, precisión, desempeño, confiabilidad, interoperabilidad y seguridad, garantizando su integración efectiva en sistemas de salud y la protección de los datos.

VI. METODOLOGÍA DE DESARROLLO

Para el correcto desarrollo del proyecto, se adoptó una metodología ágil SCRUM que se ajustó a las necesidades específicas del proyecto. Este enfoque permitió una rápida integración de funcionalidades, iteraciones frecuentes de pruebas, y una retroalimentación continua entre las partes involucradas, facilitando una mejora incremental del producto. El proyecto se dividió en varias fases clave, donde cada una abordó aspectos específicos y estableció hitos importantes. A continuación, se detalla cada fase de desarrollo:

VI-A. Fase de análisis de requerimientos

En el primer sprint, se identificaron los requerimientos funcionales y no funcionales para el proyecto basado en la investigación realizada previamente. Se consideraron tanto los aspectos técnicos como los de usabilidad, y se definieron las historias de usuario para guiar el desarrollo. Se identificaron las funcionalidades clave de la solución y se definió el cronograma a seguir durante el desarrollo en base a toda esta información.

VI-B. Fase de diseño de la solución

En esta fase, se llevó a cabo el diseño de la arquitectura del sistema basado en los requerimientos. Se crearon diagramas de arquitectura y de componentes por medio de Lucidchart, y se utilizó Figma para diseñar una interfaz de usuario intuitiva. Durante este sprint, se eligieron también las tecnologías, frameworks y bibliotecas más adecuadas para su implementación, asegurando que fueran compatibles con el modelo de reconocimiento de entidades BERT.

VI-C. Fase de desarrollo e implementación

El desarrollo del prototipo se gestionó en múltiples sprints siguiendo la metodología SCRUM, como se mencionó anteriormente. Cada tarea fue priorizada en función de las funcionalidades críticas del sistema, permitiendo iteraciones continuas sobre ellas. El código de la página web se construyó con HTML, CSS y JavaScript, y se integró el modelo BERT mediante Transformers.js para realizar el reconocimiento de entidades. Las tareas se organizaron como historias de usuario, lo que facilitó la identificación de funcionalidades clave

que aportaran valor a la solución y su posterior implementación. Además, la solución fue diseñada para anonimizar los registros médicos directamente en el navegador, eliminando la necesidad de depender de un servidor externo.

VI-D. Fase de pruebas y despliegue

Al finalizar el desarrollo, se dedicaron sprints específicos a la fase de pruebas, aplicando pruebas unitarias y funcionales para verificar que el sistema cumpliera con los requisitos. Además, se llevaron a cabo pruebas de rendimiento, usando la GPU para garantizar que la solución pudiera manejar grandes volúmenes de datos y redujera tiempos de procesamiento. Las pruebas continuas y la retroalimentación en cada sprint permitieron realizar ajustes y mejoras en el código. Posteriormente, se procedió al despliegue de la aplicación en un entorno web accesible para los usuarios mediante Vercel.

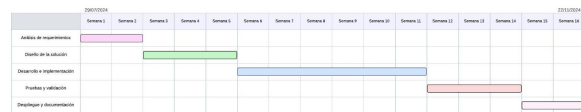


Figura 2: Cronograma del desarrollo

En el cronograma de desarrollo representado mediante un diagrama de Gantt se pueden ver en las filas las 5 fases de desarrollo definidas anteriormente, cada una con un tiempo de ejecución asignado en semanas representadas por las columnas. La distribución de tiempos por cada etapa es:

- Análisis de requerimientos: 2 semanas.
- Diseño de la solución: 3 semanas.
- Desarrollo e implementación: 6 semanas.
- Pruebas y validación: 3 semanas.
- Despliegue y documentación: 2 semanas.

VII. MARCO TEÓRICO

La anonimización de registros médicos es fundamental para cumplir con las normativas legales de privacidad y para habilitar el uso de estos datos en investigaciones científicas, sin comprometer la confidencialidad de los pacientes. Estudios recientes han subrayado la importancia de desarrollar soluciones tecnológicas avanzadas que protejan la privacidad del paciente y faciliten el acceso a grandes volúmenes de datos anonimizados para investigaciones clínicas (El Emam & Arbuckle, 2013). Sin embargo, esta es una tarea complicada debido a la necesidad de equilibrar la protección de la privacidad con la preservación de la información útil para la investigación y la toma de decisiones clínicas (Ohm, 2010). En este contexto,

el reconocimiento de entidades nombradas (NER) es una técnica crucial dentro de los procesos de anonimización de textos médicos. Esta técnica identifica automáticamente información personal que podría revelar la identidad de un paciente (Li et al., 2022). Además, el procesamiento de lenguaje natural (NLP) ha surgido como una herramienta clave para realizar esta técnica, permitiendo el análisis de grandes volúmenes de datos textuales, identificando y clasificando entidades que deben ser enmascaradas. Uno de los avances más significativos en este campo ha sido la introducción de modelos basados en Transformers, que han mejorado la capacidad de las máquinas para entender el lenguaje natural (Devlin et al., 2019).

Particularmente, los modelos basados en BERT han demostrado ser efectivos para reconocer entidades nombradas, superando a los enfoques tradicionales en cuanto a precisión y capacidad para identificar correctamente entidades clave en textos médicos. Se distingue por su capacidad de procesar texto en un contexto bidireccional, es decir, interpretando el contexto de una palabra tanto hacia adelante como hacia atrás en una oración. Esto le permite analizar no solo la palabra en sí misma, sino también cómo se relaciona con el resto de la oración para determinar si es parte de una entidad sensible con mayor precisión (Liu et al., 2019). El modelo recibe como entrada secuencias de texto que son tokenizadas en subpalabras, y procesa cada token de la secuencia generando una representación contextualizada para cada uno. Finalmente, un clasificador toma las representaciones generadas por BERT y predice etiquetas para cada token, identificando si forma parte de una entidad nombrada.

```
def attention(self, query, key, value, mask=None):
    scores = torch.matmul(query, key.transpose(-2, -1)) / math.sqrt(self.head_dim)
    if mask is not None:
        scores = scores.masked_fill(mask == 0, -1e9)
    attention_probs = nn.Softmax(dim=-1)(scores)
    context = torch.matmul(attention_probs, value)
    return context
```

Figura 3: Segmento de código de un modelo BERT

Este código pertenece al mecanismo de Self-Attention en BERT, donde se calculan las puntuaciones de atención entre todos los tokens de una secuencia. La función `attention` toma tres entradas: `query`, `key` y `value`, que son las proyecciones individuales de los tokens. Luego, calcula las probabilidades de atención, que indican cuánta "atención" poner en otros tokens al procesar un token en particular. Finalmente, se obtiene la representación contextualizada para cada uno de los tokens. Este mecanismo es esencial para tareas de

NER, ya que permite que BERT capture el contexto completo de una palabra en relación con las demás en la oración, lo que ayuda a identificar si un token forma parte de una entidad nombrada. Esta capacidad de tener en cuenta el contexto de las palabras es lo que hace tan eficiente a BERT para determinar los datos sensibles que se encuentran en un registro médico (Qu et al., 2021).

En un estudio reciente, Li et al. (2022) demostraron que, al combinar BERT con técnicas avanzadas de aprendizaje profundo y optimizaciones diseñadas específicamente para abordar el lenguaje y las complejidades del ámbito médico, se logró una mejora significativa en la precisión para identificar entidades médicas. Este enfoque permitió captar con mayor exactitud las sutilezas y variaciones del lenguaje utilizado en registros clínicos, lo que resultó en una anonimización más eficiente y segura. Complementando lo anterior, un estudio de Fan et al. (2020) aplicó BERT para la extracción de entidades en registros médicos electrónicos en mandarín, demostrando la eficacia del modelo en diferentes idiomas y configuraciones lingüísticas.

Además de los enfoques basados únicamente en aprendizaje profundo, la anonimización de datos médicos ha avanzado significativamente mediante la integración de BERT con métodos híbridos que combinan el poder del aprendizaje automático con reglas específicas del área. Por ejemplo, López-Úbeda et al. (2019) desarrollaron un enfoque que combina modelos de aprendizaje automático con reglas basadas en el conocimiento médico para mejorar la anonimización de informes clínicos en español. Este enfoque híbrido permite superar algunas de las limitaciones propias de los métodos puramente automáticos, proporcionando una mayor precisión y un mayor control en la identificación de entidades sensibles. De manera similar, Pépin y Zulkernine (2022) compararon varias herramientas de desidentificación y concluyeron que los modelos basados en BERT, especialmente cuando se entrenan con conocimientos específicos del campo médico, ofrecen una ventaja significativa en términos de precisión y flexibilidad.

Así mismo, Liu et al. (2023) presentaron una aplicación web que utiliza técnicas de aprendizaje profundo supervisado en el proceso de desidentificación de textos en registros médicos. Este estudio demostró la importancia de la supervisión humana en el uso de tecnologías avanzadas como BERT para mejorar la precisión y usabilidad de las herramientas de desidentificación. Este enfoque usado en la aplicación

permite una mayor adaptabilidad y escalabilidad en la anonimización de datos clínicos, asegurando que la información sensible se gestione de manera efectiva y segura mientras se mantiene su utilidad para la investigación. De manera similar, Juez-Hernández et al. (2023) presentaron AGORA, un innovador sistema que además de encargarse de la anonimización de documentos, también logra extraer información relevante de estos, aprovechando al máximo las capacidades avanzadas del modelo. Este sistema permite acceder a información relevante para la investigación clínica de una manera segura y eficiente, lo cual es importante cuando el equilibrio entre la protección de la privacidad y la utilidad de los datos es crucial.

En este sentido, es relevante señalar la importancia de elegir un modelo NER adecuado para lograr una solución precisa en la anonimización de registros clínicos. La capacidad del modelo para identificar correctamente las entidades dentro de los textos es clave, especialmente en contextos donde la protección de información personal es prioritaria (Colón-Ruiz & Segura-Bedmar, 2019). Según varios estudios, los modelos preentrenados como BERT han mostrado un rendimiento destacable en la tarea de anonimización, con métricas elevadas en precisión y recall, lo que los convierte en una opción sólida para esta tarea (Johnson et al., 2020; Liu et al., 2023). Por ejemplo, Martínez y Posada (2023) realizaron una evaluación exhaustiva de modelos NER preentrenados para la anonimización de textos clínicos en español, destacando que roberta-base-bne-capitel-ner-plus y ner-spanish-large obtuvieron las mejores métricas en precisión y recall, superando el 80% en ambas categorías. Además, enfatizaron la importancia de seleccionar modelos que equilibren alto rendimiento con eficiencia computacional, especialmente en entornos con recursos limitados.

Por otra parte, entrenar un modelo propio desde cero es una opción válida cuando los modelos preentrenados no se ajustan completamente a las características específicas del conjunto de datos. Aunque este enfoque puede ser más complejo y demandar mayores recursos, ofrece la ventaja de una adaptación más precisa al dominio particular. Sin embargo, este proceso es mucho más laborioso y requiere una mayor cantidad de datos etiquetados para entrenar el modelo de manera efectiva (Carmona, Conesa y Casas-Roma, 2019). Reentrenar un modelo existente puede ser una alternativa intermedia, que permita una adaptación más precisa al contexto particular del proyecto. Sin embargo, Fan, Wang y Ye (2020) advierten que, si bien reentrenar un modelo existente puede mejorar su rendimiento y es más

sencillo que entrenar un modelo desde cero, sigue demandando una gran cantidad de datos etiquetados y recursos técnicos considerables para alcanzar una alta precisión. Por ello, una evaluación cuidadosa de las necesidades y capacidades del proyecto es esencial para determinar si es preferible utilizar un modelo preentrenado, entrenar un modelo propio o reentrenar un modelo.

En resumen, la implementación de BERT en la anonimización de registros médicos representa un avance significativo en la protección de la privacidad en el ámbito de la salud. Su capacidad para entender el lenguaje natural en un contexto bidireccional, combinada con su eficacia en tareas de reconocimiento de entidades, lo convierte en una herramienta valiosa para la desidentificación de textos médicos. A medida que la tecnología continúa avanzando, es probable que el uso de BERT y otros modelos de NLP en la anonimización de datos médicos siga creciendo, mejorando tanto la protección de la privacidad como la utilidad de los datos para la investigación.

VIII. MARCO CONCEPTUAL

VIII-A. Anonimización

La anonimización es un proceso mediante el cual se elimina o enmascara la información personal que se encuentran en los datos, con el propósito de proteger la identidad de los individuos a los que dichos datos se refieren. Según Chevrier et al. (2019), la implementación efectiva de técnicas de anonimización es clave para asegurar la privacidad y el manejo ético de los datos sensibles. Por esta razón, es necesaria la anonimización de los datos para que los registros médicos puedan utilizarse para investigación sin poner en riesgo la confidencialidad de los pacientes, cumpliendo con normativas legales y manteniendo la confianza de los usuarios.

VIII-B. Registros Médicos Electrónicos (EHR)

Los registros médicos electrónicos son versiones digitales de las historias clínicas de los pacientes. La digitalización de los registros médicos ha permitido un acceso y análisis más rápidos, pero también ha aumentado los riesgos asociados con la privacidad y la seguridad [10]. Estos registros contienen información muy útil para investigaciones clínicas, pero también contienen datos personales y sensibles, por lo que su anonimización es vital para proteger la privacidad del paciente.

VIII-C. Privacidad de Datos

La privacidad de datos es un concepto que abarca la protección de la información personal frente a usos inapropiados y acceso sin autorización. Juez-Hernandez et al. (2023) resaltan la importancia de asegurar que los

datos personales permanezcan confidenciales y cumplan con los requisitos legales de protección de datos. Este principio es relevante en la salud, donde los datos médicos son muy sensibles y contienen información personal que debe tratarse cuidadosamente.

VIII-D. Reconocimiento de Entidades Nombradas (NER)

Esta es una técnica de NLP que se utiliza para identificar y clasificar nombres de personas, lugares, organizaciones y otras entidades en un texto dado. Schweter y Akbik (2017) indican que la precisión de los modelos NER, particularmente aquellos basados en BERT, ha mejorado considerablemente en los últimos años, haciendo más efectiva la protección de datos sensibles. Por esto, utilizar esta técnica en el proceso de anonimización permite identificar información sensible que necesita ser protegida para garantizar la privacidad de forma efectiva.

VIII-E. Procesamiento de Lenguaje Natural (NLP)

El NLP es un campo dentro de la inteligencia artificial que permite a las máquinas comprender y generar lenguaje humano de manera precisa. Según Liu et al. (2023), los avances en este campo han mejorado significativamente la capacidad de las máquinas para manejar tareas complejas relacionadas con textos. Esto hace posible desarrollar sistemas utilizando NLP que permitan identificar y procesar información sensible, facilitando la extracción y modificación de datos de manera eficiente y precisa.

VIII-F. Transformers

Los transformers son un tipo de arquitectura de modelo de deep learning que ha revolucionado el procesamiento del lenguaje natural y han permitido avances significativos en tareas como el reconocimiento de entidades (Wolf et al., 2020). Los transformers se caracterizan por su capacidad para manejar grandes volúmenes de texto y contextos complejos, lo que los hace especialmente eficaces en tareas como el reconocimiento de entidades nombradas (NER) y la anonimización de datos. BERT, que está basado en transformers, es un ejemplo de cómo esta tecnología ha mejorado la capacidad de las máquinas para comprender y procesar el lenguaje humano de manera eficiente.

VIII-G. BERT

Bidirectional Encoder Representations from Transformers, mejor conocido como BERT, es un modelo de procesamiento del lenguaje natural (NLP) desarrollado por Google que ha revolucionado la manera en que las máquinas comprenden y generan texto. Según Devlin et al. (2019), BERT ha demostrado un rendimiento excepcional en tareas de NLP, lo que lo convierte en una herramienta ideal para el reconocimiento de entidades. Este

modelo es capaz de comprender el contexto completo de una palabra en una frase al considerar tanto las palabras que la preceden como las que la siguen, por lo que es muy eficaz y preciso para tareas de reconocimiento de entidades.

VIII-H. Tokenización

La tokenización es una parte del procesamiento del lenguaje natural (NLP), que consiste en fragmentar un texto en unidades más pequeñas llamadas "tokens", para facilitar el reconocimiento y el análisis de este. Según Juez-Hernandez et al. (2023), la tokenización es crucial para la eficacia del reconocimiento de entidades, ya que una segmentación precisa del texto mejora la precisión del modelo en la identificación de entidades. Por esta razón, la tokenización es clave para que los modelos de inteligencia artificial puedan interpretar y procesar el lenguaje humano.

IX. ARQUITECTURA LÓGICA DE LA SOLUCIÓN

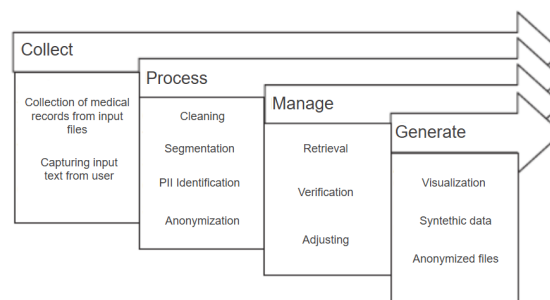


Figura 4: Ciclo de procesamiento de datos

El diagrama muestra el flujo del procesamiento de datos de la aplicación, estructurado en cuatro etapas: recolección, donde se captura texto y registros médicos en diferentes formatos; procesamiento, que incluye la limpieza, segmentación, identificación y anonimización de los datos sensibles; gestión, para verificar y ajustar los datos procesados; y generación, donde se crean nuevos registros con datos sintéticos y se permite su descarga, garantizando la seguridad y la calidad de la información.

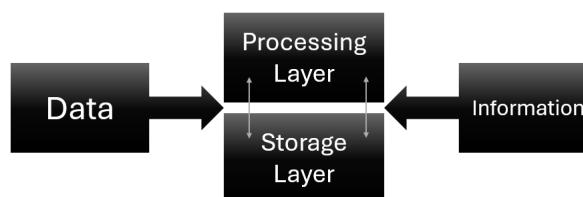


Figura 5: Unidades computacionales de procesamiento de datos

El diagrama muestra las unidades computacionales de procesamiento de datos, donde se ilustra cómo los datos son procesados para generar información útil. Los datos ingresan al sistema desde un extremo, pasan por una capa de procesamiento que interactúa temporalmente con una capa de almacenamiento mientras el usuario utiliza la página web. Esta capa de almacenamiento sirve de apoyo para realizar las operaciones necesarias en tiempo real. Finalmente, el resultado del procesamiento se transforma en información útil que es entregada al usuario. Las flechas indican el flujo continuo entre los datos, el procesamiento, el almacenamiento y la conversión en información.

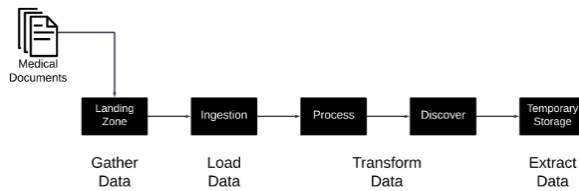


Figura 6: Flujo de procesamiento de datos

El diagrama muestra el flujo de procesamiento de datos de la aplicación, que comienza con la carga de documentos en la plataforma web, seguida de una etapa de ingestión donde los datos se preparan para el procesamiento. Luego, se aplican algoritmos de inteligencia artificial para identificar y anonimizar información personal identificable (PII). En la etapa de descubrimiento, los datos anonimizados son validados y preparados para su almacenamiento temporal, permitiendo al usuario, en la fase final, descargar los archivos procesados. Este proceso garantiza la eliminación segura de datos personales, protegiendo la privacidad mientras genera archivos útiles para análisis o investigación.

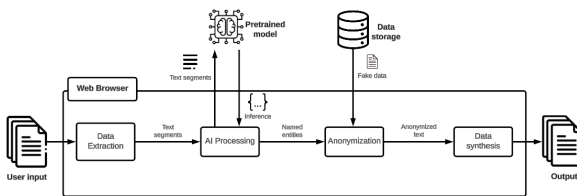


Figura 7: Arquitectura Lógica

La arquitectura lógica del proyecto se centra en la protección de los datos personales contenidos en los registros médicos, sustituyéndolos por información ficticia o "dummy". La aplicación sigue un flujo estructurado que abarca cinco etapas principales: extracción de datos, preprocesamiento, procesamiento de IA, anonimización y síntesis de datos. A lo largo de estas etapas, el sistema garantiza que los datos personales sean identificados y

sustituídos, manteniendo la coherencia en la estructura de los archivos originales.

IX-A. Extracción de datos

Esta primera etapa se encarga de tomar uno o varios archivos de registros médicos ingresados por el usuario y extraer todo el texto que contienen. Una vez se ha extraído el texto, se verifica su integridad y se fragmenta en segmentos más pequeños. Así se asegura que los datos puedan ser procesados correctamente y se mejora la precisión del análisis posterior.

IX-B. Procesamiento con IA

El sistema utiliza un modelo BERT preentrenado para realizar el reconocimiento de entidades. Cada segmento de texto es analizado por el modelo, identificando datos personales como nombres, direcciones y ubicaciones. Además de identificar estos datos, el sistema también registra su posición dentro de los segmentos de texto.

IX-C. Anonimización

En esta etapa, los datos sensibles identificados se reemplazan con información ficticia o "dummy" obtenida de una base de datos externa. Para asegurar una anonimización completa, se utilizan algoritmos adicionales, como expresiones regulares, que permiten anonimizar información que podría haber sido pasada por alto por el modelo, como fechas o identificaciones numéricas.

IX-D. Síntesis de datos

Finalmente, los segmentos de texto anonimizados se unen para recrear la estructura original de los documentos médicos. Luego, se guardan en un archivo con el mismo formato que el de entrada. De esta forma, se tiene la certeza que los archivos finales mantienen la coherencia y el formato de los originales, pero con todos los datos personales reemplazados.

X. ARQUITECTURA FÍSICA DE LA SOLUCIÓN

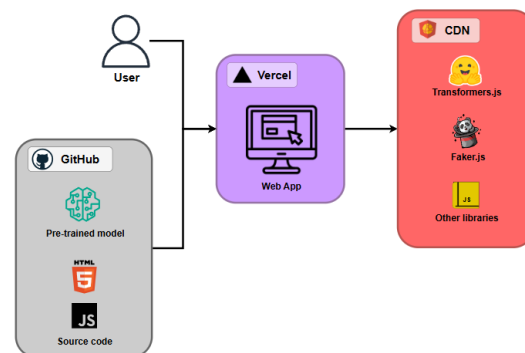


Figura 8: Arquitectura Física

El diagrama de arquitectura física de la aplicación muestra cómo se orquesta la interacción entre el usuario, la infraestructura de despliegue y las tecnologías esenciales del proyecto, ofreciendo una experiencia eficiente y sin interrupciones. El usuario accede a la aplicación a través de un navegador web, donde la interfaz está alojada mediante Vercel, una plataforma de despliegue que facilita tanto la publicación inicial como las actualizaciones continuas del front-end de la aplicación. Vercel, al ser una plataforma especializada en la gestión de aplicaciones basadas en JavaScript y frameworks web modernos, permite que el código fuente y los cambios se desplieguen de manera ágil, asegurando una alta disponibilidad del servicio.

Los archivos del proyecto, que incluyen tanto el código fuente en HTML, JavaScript, como el modelo preentrenado, se gestionan a través de GitHub. Esto facilita el control de versiones y la colaboración entre desarrolladores, permitiendo una integración continua con la plataforma de despliegue. Además, para mejorar el rendimiento de la aplicación, se emplea una Content Delivery Network (CDN) que distribuye de forma eficiente librerías como Transformers.js y Faker.js, entre otras, para que la aplicación funcione directamente desde el navegador sin necesidad de realizar llamadas a servidores externos. Al utilizar estas librerías desde la CDN, la carga de los recursos es más rápida, reduciendo la latencia y mejorando la experiencia de usuario. Así, se logra una aplicación ligera, que ejecuta operaciones de procesamiento directamente en el lado del cliente, brindando una experiencia fluida y responsiva. A continuación, se detallan las principales tecnologías empleadas en el desarrollo y funcionamiento de la aplicación:

X-A. HTML y CSS

HTML y CSS son fundamentales para la estructura y el estilo de la interfaz de la aplicación. HTML define la estructura del contenido, mientras que CSS proporciona estilos visuales que garantizan una experiencia de usuario atractiva y responsiva. Juntos, permiten crear interfaces adaptables que se ajustan a diferentes tamaños de pantalla y dispositivos, asegurando una apariencia consistente y moderna.

X-B. JavaScript

El código fuente de la aplicación está principalmente escrito en JavaScript, el lenguaje fundamental que posibilita la interacción dinámica del lado del cliente. JavaScript se encarga de manipular el DOM, integrar librerías de procesamiento de lenguaje natural y generar contenido dinámico en la interfaz del usuario. Además, permite la creación de funciones asíncronas, facilitando la carga dinámica de recursos en tiempo real.

X-C. Transformers.js

Esta librería permite la ejecución de modelos de procesamiento de lenguaje natural (NLP) directamente en el navegador. Al cargar modelos avanzados como BERT, Transformers.js disminuye la latencia al eliminar la dependencia de servidores externos para el análisis de texto. Esto asegura que las respuestas sean procesadas localmente, mejorando la privacidad del usuario y reduciendo la carga en la infraestructura del servidor.

X-D. Faker.js

Se utiliza para generar datos ficticios, simulando información útil para pruebas y demostraciones. Proporciona al sistema la capacidad de crear usuarios, textos y otros datos dinámicos sin requerir acceso a bases de datos reales, lo que agiliza el desarrollo y el testeo. Además, permite realizar pruebas de estrés y carga de la aplicación al crear escenarios con grandes volúmenes de datos de manera rápida y eficiente.

X-E. ONNX

El modelo de BERT empleado en la aplicación está compilado en formato ONNX (Open Neural Network Exchange), permitiendo su ejecución en el navegador sin la necesidad de un servidor. Esto es crucial para mantener un rendimiento rápido y eficiente al realizar tareas de procesamiento de lenguaje natural directamente en el cliente. Además, ONNX facilita la portabilidad de modelos entre diferentes frameworks de deep learning, asegurando una integración flexible con distintas plataformas.

X-F. Otras librerías

Además de Transformers.js y Faker.js, la aplicación hace uso de varias librerías adicionales de JavaScript para complementar la funcionalidad del proyecto. Estas librerías ofrecen utilidades generales como la manipulación de fechas, estructuras de datos avanzadas y animaciones, mejorando la interacción y la experiencia del usuario en la interfaz de la aplicación. Su uso optimiza la carga de la aplicación, asegurando un comportamiento más fluido y una interfaz visualmente atractiva.

X-G. Vercel

Proporciona un entorno de despliegue continuo y ágil para la aplicación web. Su integración con GitHub permite automatizar el flujo de trabajo, asegurando que cualquier actualización del código fuente se refleje de inmediato en el entorno de producción. Además, Vercel ofrece una infraestructura de edge que mejora la entrega de contenido, reduciendo la latencia y mejorando la experiencia de los usuarios globalmente.

X-H. GitHub

Almacena tanto el código fuente como los modelos preentrenados de la aplicación. Además de facilitar la colaboración entre desarrolladores, permite el control de versiones y la integración fluida con Vercel para automatizar los despliegues. Su uso garantiza la trazabilidad de cambios, lo que facilita la detección y solución de problemas en el desarrollo del proyecto.

X-I. CDN

Proporciona entrega rápida y eficiente de las librerías JavaScript que la aplicación utiliza, mejorando los tiempos de carga y la experiencia del usuario, sin importar su ubicación geográfica. Esto resulta fundamental para usuarios de diferentes regiones, asegurando que la aplicación mantenga una disponibilidad y rendimiento óptimos incluso bajo alta demanda.

XI. PROTOTIPO

A continuación se detallarán las funcionalidades clave del prototipo, así como los métodos implementados para garantizar que esta solución responda a las necesidades actuales de seguridad en el manejo de datos médicos:

XI-A. Selección del modelo

Para este proyecto, la selección del modelo de reconocimiento de entidades nombradas (NER) es fundamental, ya que el objetivo es identificar y anonimizar de manera precisa y eficiente información sensible en registros médicos. Se decidió utilizar el modelo “roberta-base-bne-capitel-ner-plus” en base a la investigación realizada por D. Martínez y J. Posada (2023) quienes evaluaron un conjunto de modelos de NER en español y encontraron que este modelo, tras un proceso de relajación de límites, lograba un desempeño excepcional en la extracción de entidades clave en el contexto clínico. En particular, el modelo alcanzó métricas sobresalientes con una precisión de 0.82, una exhaustividad de 0.87 y una puntuación F1 de 0.84, ubicándolo entre los mejores para la tarea de anonimización de datos en español. Además, al ser un modelo ligero también presentaba un excelente rendimiento con tiempo de respuesta corto. De esta forma, este modelo se ubicó como la mejor opción con un equilibrio entre precisión y velocidad.

Debido a que se está desplegando con la librería Transformers.js, un marco que permite implementar modelos de inteligencia artificial directamente en el navegador, fue necesario compilar el modelo al formato ONNX. Este formato permite desplegar el modelo directamente en el navegador de manera rápida y sin comprometer la privacidad del usuario, ya que el procesamiento se realiza de manera local. Esto no solo

asegura la velocidad de ejecución, sino también un control sobre los datos procesados, manteniendo la integridad y la privacidad de la información.

XI-B. Arquitectura y funcionamiento del prototipo

El prototipo desarrollado para la anonimización de registros médicos sigue una arquitectura lógica compuesta por una serie de scripts que trabajan en conjunto para extraer, procesar y anonimizar información sensible. Esta sección detalla la estructura y el propósito de cada uno de los scripts, explicando su funcionamiento y cómo se relacionan entre sí para cumplir con el objetivo de anonimización eficiente y segura.

XI-B1. index.js: El archivo `index.js` actúa como el punto de entrada principal de la aplicación. Este script se encarga de inicializar la interfaz de usuario y de gestionar la interacción entre el usuario y las funciones de procesamiento de texto. Incluye la lógica para manejar eventos de carga de archivos, la visualización de resultados y la llamada a funciones clave como `runNER` para procesar el texto con el modelo de NER. Además, coordina la visualización de los textos originales y anonimizados en la página web, permitiendo al usuario descargar los resultados en diferentes formatos, como archivos `.txt` o `.docx`. Este archivo es fundamental para la experiencia de usuario, ya que conecta la lógica del procesamiento de datos con la interfaz de la aplicación.

```
document.getElementById('file-input').addEventListener('change', function() {
  var fileInput = document.getElementById('file-input');
  var fileCount = document.getElementById('file-count');
  document.getElementById('downloadSingleBtn').style.display = 'none';
  usedFileNames.clear();
  usedFileTypes.clear();
  if (fileInput.files.length === 0) {
    fileCount.textContent = 'No hay archivos seleccionados';
    usingFile = false;
  } else if (fileInput.files.length === 1) {
    fileCount.textContent = '1 archivo seleccionado';
    usingFile = false;
  } else {
    fileCount.textContent = fileInput.files.length + ' archivos seleccionados';
    usingFile = true;
  }
});
```

Figura 9: Evento para subida de archivos

XI-B2. dataExtraction.js: Este script es responsable de la extracción de texto desde archivos, en particular documentos PDF, utilizando la librería `pdfjsLib`. La función principal es `extractTextFromPDF`, que itera a través de las páginas de un archivo PDF y extrae el contenido manteniendo la estructura del texto, lo que permite identificar de manera precisa los saltos de línea y otras características del formato original. Esta función es fundamental para garantizar que el texto procesado por

el sistema preserve su estructura y pueda ser analizado correctamente por el modelo de NER. Esta función es clave para lograr manejar distintos formatos, al garantizar que el procesamiento de archivos PDF sea posible en esta herramienta.

```

async function extractTextFromPDF(arrayBuffer) {
  const loadingTask = pdfjsLib.getDocument({ data: arrayBuffer });
  const pdf = await loadingTask.promise;
  let text = '';

  for (let i = 1; i <= pdf.numPages; i++) {
    const page = await pdf.getPage(i);
    const textContent = await page.getTextContent();
    let pageText = '';

    textContent.items.forEach((item, index) => {
      pageText += item.str;
      if (index < textContent.items.length - 1) {
        const currentY = item.transform[5];
        const nextY = textContent.items[index + 1].transform[5];
        if (Math.abs(currentY - nextY) > 10) {
          pageText += '\n';
        }
      }
    });

    text += pageText + '\n';
  }

  return text;
}

```

Figura 10: Función extractTextFromPDF

XI-B3. aiProcessing.js: Este script gestiona la interacción con el modelo de reconocimiento de entidades nombradas (NER). La función runNER recibe el texto de entrada, lo divide en segmentos y lo procesa usando el modelo seleccionado. Utilizando Transformers.js, se carga y ejecuta el modelo de manera local en el navegador, asegurando un procesamiento rápido y privado. La función también llama a otras funciones auxiliares, como cleanEntities y filterEntities, que limpian y combinan las entidades detectadas, eliminando duplicados y uniendo tokens adyacentes.

Como se observa en las imágenes, runNER recibe el texto y lo divide en segmentos, los cuales son analizados por el modelo para identificar entidades como nombres de personas, organizaciones y ubicaciones. Por otro lado, cleanEntities limpia las entidades y combina aquellas que tengan un índice consecutivo para formar una sola entidad, garantizando utilizar entidades correctas para el modelo. Mientras que la función auxiliar filterEntities garantiza que las entidades sean correctas al descartar entidades con una puntuación menor a 0.6.

```

async function runNER(inputText, mode, filename = '') {
  const resultDiv = document.getElementById('result');
  resultDiv.classList.remove('hidden');

  if (!inputText) {
    resultDiv.textContent = 'Por favor, ingrese algún texto.';
    return;
  }

  const loadingLabel = document.createElement('p');
  loadingLabel.id = `file-${filename}`;
  loadingLabel.textContent = `Analizando ${filename}...`;
  resultDiv.appendChild(loadingLabel);

  try {
    const segments = splitText(inputText);
    let cleanedEntities = [];
    let replacedTextLines = [];
    for (const segment of segments) {
      const entities = await pipe(segment);
      const cleanedSegmentEntities = cleanEntities(entities);
      const filteredEntities = filterEntities(cleanedSegmentEntities);
      cleanedEntities = cleanedEntities.concat(filteredEntities);
      replacedTextLines.push(replaceEntities(segment, filteredEntities, mode));
    }
    let replacedText = replacedTextLines.join('\n');
    replacedText = secondaryReplacements(replacedText, mode);

    loadingLabel.remove();
    await displayResults(inputText, replacedText, filename);
    // Si solo se analiza un archivo, se muestra el botón de descarga individual
    if (filename && document.getElementById('file-input').files.length === 1) {
      document.getElementById('downloadSingleBtn').style.display = 'block';
      document.getElementById('downloadZipBtn').style.display = 'none';
    } else {
      document.getElementById('downloadZipBtn').style.display = 'block';
      document.getElementById('downloadSingleBtn').style.display = 'none';
    }
    document.getElementById('clearResultsBtn').classList.remove('hidden');
  } catch (error) {
    resultDiv.innerHTML += `<p>Error al procesar el texto ${filename}: ${error}</p>`;
  }
}

```

Figura 11: Función runNER

```

function cleanEntities(entities) {
  const cleanedEntities = [];
  let currentEntity = null;

  entities.forEach((entity) => {
    // Eliminar espacios al principio y al final de la palabra
    entity.word = entity.word.trim();
    if (currentEntity && entity.index === currentEntity.index + 1) {
      if (entity.word.startsWith('#')) {
        currentEntity.word += entity.word.replace('#', '');
      } else {
        currentEntity.word += entity.word;
      }
      currentEntity.index = entity.index;
    } else {
      if (currentEntity) {
        cleanedEntities.push(currentEntity);
      }
      currentEntity = { ...entity };
    }
  });

  if (currentEntity) {
    cleanedEntities.push(currentEntity);
  }

  return cleanedEntities;
}

```

Figura 12: Función cleanEntities

```

// Función para filtrar las entidades con una puntuación menor a 0.6
function filterEntities(entities) {
  return entities.filter(entity => entity.score >= 0.6);
}

```

Figura 13: Función filterEntities

XI-B4. dataAnonymization.js: Este script gestiona el reemplazo de las entidades detectadas en el texto por datos anonimizados. La función `replaceEntities` toma el texto de entrada y la lista de entidades detectadas y las reemplaza utilizando marcadores genéricos o datos falsos generados por la librería `faker`. Dependiendo del modo de anonimización seleccionado, los reemplazos pueden ser realistas (modo `advanced`) o simples etiquetas de marcadores (modo `generic`). La función ordena las entidades por longitud para evitar reemplazos parciales y garantiza que las palabras se sustituyan de forma precisa y coherente.

```
function replaceEntities(text, entities, mode) {
  let replacedText = text;

  // Ordenar entidades de la más larga a la más corta para evitar reemplazos parciales
  entities.sort((a, b) => b.word.length - a.word.length);

  entities.forEach(entity => {
    let replacement;

    if (mode === 'advanced') {
      if (entity.entity.includes('PER')) {
        replacement = faker.person.firstName() + ' ' + faker.person.lastName();
      } else if (entity.entity.includes('LOC')) {
        replacement = /\/d/.test(entity.word) ? faker.location.streetAddress(true) : faker.location.country();
      } else if (entity.entity.includes('ORG')) {
        replacement = faker.company.name();
      } else {
        //Aquí se puede configurar el manejo para entidades como MISC u OTH
      }
    } else if (mode === 'generic') {
      if (entity.entity.includes('PER')) {
        replacement = '[persona]';
      } else if (entity.entity.includes('LOC')) {
        replacement = /\/d/.test(entity.word) ? '[dirección]' : '[lugar]';
      } else if (entity.entity.includes('ORG')) {
        replacement = '[organización]';
      } else {
        //Aquí se puede configurar el manejo para entidades como MISC u OTH
      }
    }

    const regex = new RegExp(`\\b${entity.word.split('').join('\\s*')}\\b`, 'g');
    replacedText = replacedText.replace(regex, replacement);
  });

  return replacedText;
}
```

Figura 14: Función `replaceEntities`

XI-B5. dataSynthesis.js: Este script maneja la presentación de los resultados y la generación de archivos de salida. La función `displayResults` muestra el texto original y el texto anonimizado en una interfaz que permite al usuario realizar ediciones en tiempo real. Las modificaciones realizadas por el usuario se actualizan en un mapa de texto que se prepara para la descarga. Además, las funciones `updateZipFile` y `generateZip` permiten que el usuario descargue los resultados anonimizados en formatos TXT, DOCX y PDF, generando un archivo comprimido que contiene todos los documentos procesados.

El flujo de trabajo garantiza que la extracción, detección y anonimización de entidades, así como la visualización de resultados, se realicen de manera integrada y eficiente, proporcionando al usuario un sistema completo de anonimización de datos.

XI-C. Usabilidad del Prototipo

La aplicación web prototipo, Safe Records, se presenta como una solución para la anonimización de registros médicos. Diseñado para operar completamente en el navegador, ofrece un conjunto importante de funcionalidades que facilitan la protección de datos sensibles en documentos médicos, manteniendo una interfaz simple y concreta. A continuación, se detallan

las principales capacidades del prototipo y cómo el usuario puede interactuar con ellas.

XI-C1. Carga de archivos: La página de inicio del prototipo permite al usuario cargar uno o más documentos médicos en formatos como PDF, DOCX y TXT. La interfaz es clara y cuenta con un botón principal para la selección de archivos desde el sistema local y, además, se muestra un indicador visual de la cantidad de archivos cargados. De igual manera, el usuario puede escribir manualmente un texto a ser anonimizado.

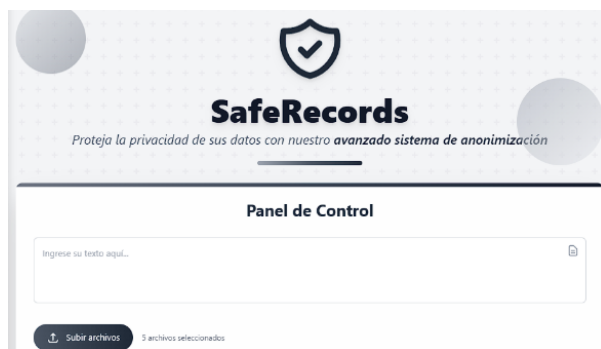


Figura 15: Panel de carga de archivos y texto

XI-C2. Proceso de anonimización: Una vez cargados los archivos, el prototipo permite al usuario cargar desde HuggingFace el modelo que desea usar colocando el nombre de su repositorio o dejar el modelo por defecto (SantiMB/roberta-base-bne-capitelner-plus-ONNX), para cambiarlo se debe presionar el botón de Aplicar, y posteriormente se muestra un indicador visual que indica la carga del modelo.

Adicionalmente, el usuario puede configurar el modo de anonimización entre dos opciones de un menú desplegable: genérico y avanzado. La anonimización genérica sustituye las entidades sensibles por etiquetas genéricas como [PERSONA] o [LUGAR], mientras que la avanzada las reemplaza por datos dummy que sean coherentes y mantengan el sentido del informe. Una vez seleccionada la configuración de anonimización deseada, solamente es necesario presionar el botón “Anonimizar” para iniciar con el proceso.

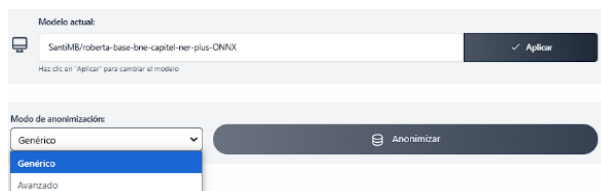


Figura 16: Panel de configuración de anonimización

XI-C3. Visualización, edición y descarga de resultados: Una vez finalizada la anonimización de los registros médicos, los resultados del proceso se presentan al usuario en desplegables con dos paneles paralelos para cada texto cargado. El primer panel muestra el texto original cargado, y la segunda muestra el contenido procesado con la información sensible sustituida. Además, la interfaz permite al usuario realizar ediciones manuales en tiempo real para ajustar el texto anonimizado si es necesario.

Una vez verificado todos los resultados, el usuario puede descargar fácilmente los textos anonimizados presionando el botón “Descargar Archivos Anonimizados”, todos los textos estarán comprimidos en un .zip, cada uno con el formato del texto original (PDF, DOCX o TXT). En caso de no querer descargar los resultados, el usuario puede presionar el botón “Borrar resultados” para dejar de mostrarlos.

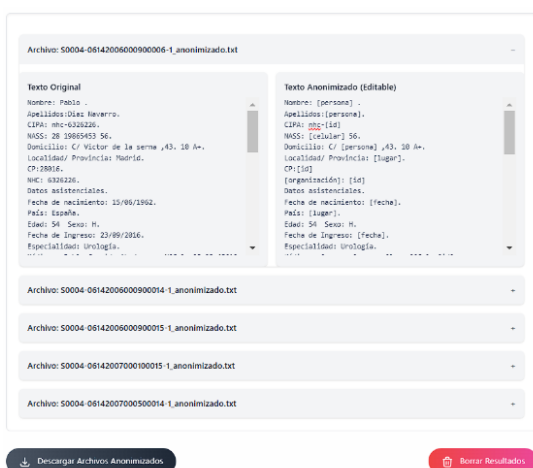


Figura 17: Panel de resultados

El prototipo no solo simplifica el proceso de anonimización de registros médicos, sino que también ofrece a los usuarios un control total sobre los resultados obtenidos, asegurando la privacidad y precisión en el manejo de datos sensibles.

XII. TABLA DE EVALUACIÓN

Característica	Definición o descripción	1	2	3	4	5
Understandability	¿Facil de comprender?					5
	¿Documentación de usuario completa, apropiada y bien estructurada?					5
Documentation	¿Facil de consultar en un sistema compatible? (Close-Open)				4	
	¿Facil de instalar en un sistema compatible?				4	
Buildability	¿Facil de aprender a usar sus funciones?					5
	¿La identidad del proyecto / software es clara y única?					5
Learnability	¿Es facil ver quien posee el proyecto / software?					
	¿Adopción de la licencia apropiada?					
Identity	¿Facil de entender como se ejecuta el proyecto y como se gestiona el desarrollo del software?			3		
	¿Evidencia de comunidad actual / futura?					
Governance	¿Evidencia de capacidad de desarrollo actual / futura?					
	¿Facil de probar la corrección de las funciones caja negra?				4	
Community	¿Utilizable en multiples plataformas?					5
	¿Evidencia de soporte para desarrolladores actuales / futuros?					
Accessibility	¿Facil de entender a nivel fuente?			3		
	¿Facil de modificar y aportar cambios a los desarrolladores?					5
Portability	¿Facil de desarrollo actual / futuro?					
	¿Interoperable con otro software requerido / relaciona					

Figura 18: Tabla de validación del prototipo

Esta tabla es una representación numérica del cumplimiento de ciertos factores a evaluar del prototipo, realizado por un grupo de control que desarrollaban su propio proyecto de investigación en paralelo. Esta tabla toma en cuenta campos como la comprensión, documentación, portabilidad, modularidad, y demás aspectos del prototipo; facilitando la identificación de sus puntos fuertes y sus aspectos a mejorar.

XIII. CONCLUSIONES Y RESULTADOS

El proyecto cumplió satisfactoriamente con los objetivos planteados, demostrando la viabilidad de una solución basada en inteligencia artificial para la anonimización de registros médicos. Utilizando modelos de reconocimiento de entidades como BERT, se logró identificar y reemplazar información sensible de manera precisa, respetando los principios de privacidad y preservando la utilidad de los datos anonimizados para su uso en investigaciones científicas. Esta herramienta permite proteger la privacidad de los usuarios mientras mantiene la integridad de los datos anonimizados, un aspecto crucial en el contexto de la investigación médica.

La implementación del prototipo integró exitosamente un modelo preentrenado en una aplicación web accesible, cumpliendo con el objetivo de ofrecer una solución práctica, segura y eficiente. Esto fue posible gracias al despliegue de tecnologías modernas como Transformers.js y ONNX, que garantizan la ejecución local del procesamiento en el navegador, eliminando la necesidad de transferir datos a servidores externos. Este enfoque no solo asegura la privacidad, sino que también mejora la accesibilidad al proporcionar una herramienta liviana, rápida y confiable.

El corpus público notas médicas en español, MEDDOCAN, fue utilizado para validar la solución, asegurando su capacidad para identificar y anonimizar información sensible con precisión. Las pruebas confirmaron que la aplicación puede procesar hasta 20 archivos simultáneamente, manteniendo un rendimiento estable y tiempos de respuesta adecuados para los usuarios. La precisión promedio alcanzada fue del 74 %, lo cual es un buen resultado, aunque mejorable.

En términos de usabilidad, la interfaz de SafeRecords fue diseñada para ser intuitiva, permitiendo a los usuarios cargar documentos en múltiples formatos, visualizar los textos anonimizados y descargarlos en un paquete comprimido. Además, la herramienta incluye opciones de anonimización avanzada, que reemplazan entidades sensibles con datos ficticios contextualmente coherentes. Estas características destacan el valor de la aplicación como una solución integral para proteger la

privacidad en documentos médicos.

El desarrollo del proyecto se llevó a cabo bajo una metodología ágil, facilitando el cumplimiento de hitos clave desde la revisión bibliográfica y el diseño de la arquitectura, hasta la validación del prototipo. Las pruebas realizadas confirmaron altos estándares de precisión y rendimiento, gracias a la robustez del diseño técnico y la integración eficiente de tecnologías innovadoras. Esto asegura que la herramienta no solo cumple con las expectativas actuales, sino que también establece un estándar en la automatización de procesos de anonimización médica.

Al final, se logró desarrollar una solución tecnológica que responde a las necesidades del sector salud, garantizando un equilibrio entre la seguridad de los datos y su utilidad. Este proyecto no solo ofrece una herramienta efectiva para proteger la privacidad de los pacientes, sino que también sienta un precedente en la aplicación de inteligencia artificial en el ámbito médico, abriendo camino para futuras investigaciones y desarrollos en esta área.

REFERENCIAS

- [1] F. Carmona, J. Conesa, and J. Casas-Roma, "Towards the analysis of how anonymization affects the usefulness of health data in the context of machine learning," *Semantic Scholar*, 2019. [Online]. Available: <https://www.semanticscholar.org/paper/Towards-the-Analysis-of-How-Anonymization-Affects-Carmona-Conesa/fe4324802561d5b8b54ba059cc9e6aca269d0cbb>. [Accessed: 23-Jul-2024].
- [2] Q. Wang and H. E, "A BERT-based named entity recognition in Chinese electronic medical records," in *Proc. 2020 9th Int. Conf. Comput. Pattern Recognition*, 2021, pp. 1165–1169, doi: 10.1145/3436369.3436390.
- [3] I. Pépin and F. Zulkernine, "A comparative study of de-identification tools to apply to free-text clinical notes," in *Proc. 32nd Annu. Int. Conf. Comput. Sci. Softw. Eng.*, 2022, doi: 10.5555/3566055.3566077.
- [4] H. Fan, D. Wang, and S. Ye, "Named entity extraction for Chinese electronic medical records," in *Proc. 2019 3rd Int. Conf. Comput. Sci. Artif. Intell.*, 2020, pp. 278–282, doi: 10.1145/3374587.3374612.
- [5] A. E. W. Johnson, L. Bulgarelli, and T. J. Pollard, "Deidentification of free-text medical records using pre-trained bidirectional transformers," in *Proc. ACM Conf. Health, Infer. Learn.*, 2020, pp. 214–221, doi: 10.1145/3368555.3384455.
- [6] Z. Li, H. Yun, Z. Guo, and J. Qi, "Medical named entity recognition based on multi-feature fusion of BERT," in *Proc. 4th Int. Conf. Big Data Technol.*, 2022, pp. 86–91, doi: 10.1145/3490322.3490336.
- [7] Y. Wang and X. Zhang, "Research on named entity recognition for Chinese medical case reports," in *Proc. 2023 4th Int. Symp. Artif. Intell. Med. Sci.*, 2024, pp. 1165–1169, doi: 10.1145/3644116.3644314.
- [8] A. Kovacevic, B. Bašaragin, N. Milosevic, and G. Nenađić, "De-identification of clinical free text using natural language processing: A systematic review of current approaches," *Artif. Intell. Med.*, vol. 151, p. 102845, 2023, doi: 10.1016/j.artmed.2023.102845.
- [9] T. Ahmed, M. M. A. Aziz, N. Mohammed, and X. Jiang, "Privacy preserving neural networks for electronic health records de-identification," in *Proc. 12th ACM Conf. Bioinform., Comput. Biol., Health Inform.*, 2021, Art. no. 8, pp. 1–6, doi: 10.1145/3459930.3469555.
- [10] J. Gardner, L. Xiong, F. Wang, A. Post, J. Saltz, and T. Grandison, "An evaluation of feature sets and sampling techniques for de-identification of medical records," in *Proc. 1st ACM Int. Health Informat. Symp.*, 2010, pp. 183–190, doi: 10.1145/1882992.1883019.
- [11] C. Qu, W. Kong, L. Yang, M. Zhang, M. Bendersky, and M. Najork, "Natural language understanding with privacy-preserving BERT," in *Proc. 30th ACM Int. Conf. Inform. Knowl. Manag.*, 2021, pp. 1488–1497, doi: 10.1145/3459637.3482281.
- [12] P. Báez, F. Bravo-Marquez, J. Dunstan, M. Rojas, and F. Villena, "Automatic extraction of nested entities in clinical referrals in Spanish," *ACM Trans. Comput. Healthc.*, vol. 3, no. 3, Art. no. 28, p. 22, 2022, doi: 10.1145/3498324.
- [13] R. Juez-Hernandez, L. Quijano-Sánchez, F. Liberatore, and J. Gómez, "AGORA: An intelligent system for the anonymization, information extraction, and automatic mapping of sensitive documents," *Appl. Soft Comput.*, vol. 145, p. 110540, 2023, doi: 10.1016/j.asoc.2023.110540.
- [14] E. Alvarez-Mellado and C. Lignos, "Detecting unassimilated borrowings in Spanish: An annotated corpus and approaches to modeling," in *Proc. 60th Annu. Meeting Assoc. Comput. Linguist.*, 2022, pp. 3868–3888, doi: 10.18653/v1/2022.acl-long.268.
- [15] J. Mao and W. Liu, "Hadoken: A BERT-CRF model for medical document anonymization," *IberLEF@SEPLN*, 2019.
- [16] V. Cotik, F. M. Luque, and J. M. Pérez, "Window Classifiers and Conditional Random Fields for Medical Report De-Identification," 2019.
- [17] S. Schweter and A. Akbik, "FLERT: Document-level features for named entity recognition," 2021.
- [18] P. López-Úbeda, M. C. Díaz-Galiano, L. A. López, and M. T. Valdivia, "Anonymization of Clinical Reports in Spanish: a Hybrid Method Based on Machine Learning and Rules," 2019.
- [19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2019.
- [20] M. Marimon, A. Gonzalez-Agirre, A. Intxaurre, J. A. L. Martin, and M. Villegas, "Automatic de-identification of medical texts in Spanish: The MEDDOCAN track, corpus, guidelines, methods, and evaluation of results," 2019.
- [21] A. Sepas, A. H. Bangash, O. Alraoui, K. El Emam, and A. El-Hussuna, "Algorithms to anonymize structured medical and healthcare data: A systematic review," *Front. Bioinform.*, vol. 2, p. 984807, 2022, doi: 10.3389/fbinf.2022.984807.
- [22] C. Colón-Ruiz and I. Segura-Bedmar, "Protected Health Information Recognition by BiLSTM-CRF," 2019.
- [23] L. Liu, O. Perez-Concha, A. Nguyen, V. Bennett, B. Blake, B. Gallego, and L. Jorm, "Web-Based Application Based on Human-in-the-Loop Deep Learning for Deidentifying Free-Text Data in Electronic Medical Records: Development and Usability Study," *Interact. J. Med. Res.*, vol. 12, p. e46322, 2023, doi: 10.2196/46322.
- [24] R. Chevrier, V. Foufi, C. Gaudet-Blavignac, A. Robert, and C. Lovis, "Use and Understanding of Anonymization and De-Identification in the Biomedical Literature: Scoping Review," *J. Med. Internet Res.*, vol. 21, no. 5, p. e13484, 2019, doi: 10.2196/13484.
- [25] B. Negash, A. Katz, C. J. Neilson, M. Moni, M. Nesca, A. Singer, and J. E. Enns, "De-identification of free text data containing personal health information: a scoping review of reviews," *Int. J. Popul. Data Sci.*, vol. 8, no. 1, p. 2153, 2023, doi: 10.23889/ijpds.v8i1.2153.
- [26] A. García-Pablos, N. Pérez, and M. Cuadros, "Sensitive data detection and classification in Spanish clinical text: Experiments with BERT," *ArXiv*, 2020, doi: 10.48550/arXiv.2003.03106.
- [27] J. Liu, S. Gupta, A. Chen, C. K. Wang, P. Mishra, H. J. Dai, Z. S. Wong, and J. Jonnagaddala, "OpenDeID pipeline for

unstructured electronic health record text notes based on rules and transformers: Deidentification algorithm development and validation study,"*J. Med. Internet Res.*, vol. 25, p. e48145, 2023, doi: 10.2196/48145.

- [28] L. Liu, O. Perez-Concha, A. Nguyen, V. Bennett, and L. Jorm, "De-identifying Australian hospital discharge summaries: An end-to-end framework using ensemble of deep learning models,"*J. Biomed. Inform.*, vol. 135, p. 104215, 2022, doi: 10.1016/j.jbi.2022.104215.
- [29] A. Ahmed, A. Abbasi, and C. Eickhoff, "Benchmarking Modern Named Entity Recognition Techniques for Free-text Health Record Deidentification," in *AMIA Joint Summits Transl. Sci. Proc.*, 2021, pp. 102–111.
- [30] T. Wolf, et al., "HuggingFace's transformers: State-of-the-art natural language processing," 2020.
- [31] I. Pérez-Díez, R. Pérez-Moraga, A. López-Cerdán, J. M. Salinas-Serrano, and M. la Iglesia-Vayá, "De-identifying Spanish medical texts—Named entity recognition applied to radiology reports,"*J. Biomed. Semant.*, vol. 12, no. 1, p. 6, 2021, doi: 10.1186/s13326-021-00237-2.
- [32] D. Martínez y J. Posada, "Analysis of Pre-trained Language Models in Text Classification for Use in Spanish Medical Records Anonymization," 2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2023, pp. 214-221. doi: 10.1109/BIBM57415.2023.9999999.