UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR
DE INGENIEROS DE TELECOMUNICACIÓN

GRADO EN INGENIERÍA DE TECNOLOGÍAS Y SERVICIOS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

Desarrollo de un agente conversacional para apoyo en diagnóstico clínico basado en modelos de lenguaje natural LLMs

Mónica Ferrer Gómez-Cano

2025

Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación

TRABAJO DE FIN DE GRADO

Título: Desarrollo de un agente conversacional para apoyo en diagnóstico clínico basado en modelos de lenguaje natural LLMs

Autor: Mónica Ferrer Gómez-Cano

Tutor: Dr. Alberto Belmonte Hernández

Departamento: Señales, Sistemas y Radiocomunicaciones (SSR)

MIEMBROS DEL TRIBUNAL

Presidente: ----------

Vocal: ----------

Secretario: ----------

Suplente: ----------

Los miembros del tribunal arriba nombrados acuerdan otorgar la calificación de:

Fecha de lectura: ----------

UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE TELECOMUNICACIÓN

GRADO EN INGENIERÍA DE TECNOLOGÍAS Y SERVICIOS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

Desarrollo de un agente conversacional para apoyo en diagnóstico clínico basado en modelos de lenguaje natural LLMs

Mónica Ferrer Gómez-Cano

2025

Resumen

El presente Trabajo de Fin de Grado explora la unión de la inteligencia artificial y la medicina, abordando uno de los grandes retos actuales: cómo transformar la creciente cantidad de información médica en conocimiento útil para el diagnóstico.

En un contexto donde la literatura médica se duplica cada 73 días y los profesionales disponen de apenas 6 a 10 minutos por paciente, la necesidad de herramientas inteligentes de apoyo al diagnóstico se vuelve crítica.

Esta investigación compara el potencial de tres enfoques de procesamiento del lenguaje natural aplicados al diagnóstico médico:

1. **Modelos generalistas sin entrenamiento específico**, evaluando su capacidad para comprender y responder a consultas clínicas complejas.
2. **Fine-tuning**, que adapta un modelo con conocimientos generales al dominio médico mediante exposición a miles de casos clínicos, síntomas y tratamientos.
3. **Retrieval-Augmented Generation (RAG)**, un sistema que combina generación de texto con recuperación de información médica actualizada, permitiendo respuestas fundamentadas en evidencia reciente sin necesidad de entrenar el modelo.

La metodología desarrollada incluye la creación de un conjunto de datos clínicos mediante técnicas avanzadas de *web scraping*, extrayendo información verificada de fuentes médicas reconocidas como Mayo Clinic y RxList, generando un dataset estructurado de 1.150 enfermedades y 3.929 fármacos, estableciendo relaciones semánticas complejas entre síntomas, diagnósticos y tratamientos.

La evaluación se basó en métricas avanzadas que van más allá de la coherencia textual, incluyendo precisión diagnóstica, contextualización clínica, eficiencia temporal y fiabilidad. Además, se utilizó la herramienta RAGAS para medir aspectos como la relevancia del contexto recuperado y la precisión de las respuestas al contenido médico.

El proyecto culmina con el diseño de un agente conversacional clínico que integra el enfoque más eficaz, capaz de analizar síntomas, generar diagnósticos diferenciales y justificar sus respuestas con evidencia científica. Más allá de lo técnico, esta propuesta contribuye al desarrollo de una medicina más personalizada, donde la inteligencia artificial complementa y amplifica el juicio clínico humano.

Palabras clave

Inteligencia artificial, Modelos de lenguaje natural (LLM), Diagnóstico clínico, Agente conversacional, Fine-tuning, RAG, Web scraping, PLN, transformers, embeddings

Summary

This Final Degree Project explores the union between artificial intelligence and clinical medicine, addressing one of today's most pressing challenges: how to transform the growing volume of medical information into useful knowledge for diagnosis.

In a context where medical literature doubles every 73 days and healthcare professionals have only 6 to 10 minutes per patient, the need for intelligent diagnostic support tools becomes critical.

This research compares the potential of three natural language processing approaches applied to medical diagnosis:

1. **General-purpose language models without domain-specific training**, assessing their ability to understand and respond to complex clinical queries.
2. **Fine-tuning**, which adapts a general-purpose model to the medical domain by exposing it to thousands of clinical cases, symptoms, and treatments.
3. **Retrieval-Augmented Generation (RAG)**, a system that combines text generation with real-time retrieval of updated medical information, enabling responses grounded in recent evidence without retraining the model.

The methodology includes the creation of a clinical dataset using advanced *web scraping* techniques, extracting verified information from reputable medical sources such as Mayo Clinic and RxList. This process resulted in a structured dataset of 1,150 diseases and 3,929 drugs, establishing complex semantic relationships between symptoms, diagnoses, and treatments.

The evaluation was based on advanced metrics that go beyond textual coherence, including diagnostic accuracy, clinical contextualization, response efficiency, and reliability. Additionally, the RAGAS tool was used to assess aspects such as the relevance of retrieved context and the factual consistency of the generated responses.

The project concludes with the design of a clinical conversational agent that integrates the most effective approach, capable of analyzing symptoms, generating differential diagnoses, and justifying its suggestions with scientific evidence. Beyond its technical contributions, this work supports the development of truly personalized medicine, where artificial intelligence enhances—rather than replaces—clinical judgment.

Keywords

Artificial intelligence, Large Language Models (LLMs), Clinical diagnosis, Conversational agent, Fine-tuning, RAG, Web scraping, NLP, Transformers, Embeddings

Índice

Listado de figuras

Listado de tablas

Glosario

A

API Application Programming Interface - Interfaz que permite la comunicación entre diferentes sistemas de software, facilitando el intercambio de datos y funcionalidades.

ATENCIÓN PRIMARIA Primer nivel de contacto de los pacientes con el sistema sanitario, donde se resuelve aproximadamente el 80 % de los problemas de salud.

ATTENTION MECHANISM Mecanismo de atención - Técnica que permite a los modelos de lenguaje enfocarse en partes relevantes de la secuencia de entrada, mejorando la comprensión contextual.

B

BACKPROPAGATION Algoritmo fundamental para calcular gradientes en redes neuronales, propagando el error hacia atrás por las capas para ajustar los pesos.

BATCH SIZE Tamaño de lote - Número de muestras procesadas en paralelo antes de actualizar los parámetros del modelo durante el entrenamiento.

BEAUTIFULSOUP Librería de Python especializada en extraer datos de archivos HTML y XML, ampliamente utilizada en web scraping.

BIOBERT Modelo BERT especializado en textos biomédicos, entrenado sobre corpus de literatura científica médica para mejorar la comprensión de terminología clínica.

BIOGPT Modelo de lenguaje especializado en el dominio biomédico, basado en la arquitectura GPT y entrenado específicamente con literatura médica.

BLEU Bilingual Evaluation Understudy - Métrica estándar para evaluar la calidad de texto generado automáticamente comparándolo con referencias humanas.

C

CHECKPOINT Punto de guardado durante el entrenamiento que preserva el estado completo de los parámetros del modelo, permitiendo reanudar o evaluar diferentes etapas.

CHUNKS Fragmentos de texto de tamaño fijo utilizados en sistemas RAG para facilitar la recuperación eficiente de información relevante.

CNN Convolutional Neural Networks (Redes Neuronales Convolucionales) - Arquitectura especializada en procesar datos con estructura espacial como imágenes.

CORPUS Colección estructurada y organizada de textos utilizados para entrenamiento, análisis lingüístico o evaluación de modelos de lenguaje.

COSINE SIMILARITY Similitud del coseno - Métrica que mide la similitud entre dos vectores basándose en el ángulo entre ellos, muy utilizada en análisis semántico.

CROSS-VALIDATION Validación cruzada - Técnica estadística para evaluar la capacidad de generalización de un modelo dividiendo los datos en múltiples conjuntos.

CSV Comma-Separated Values - Formato de archivo estándar para almacenar datos tabulares separados por comas, fácil de procesar y compartir.

CSS Cascading Style Sheets – lenguaje de hojas de estilo en cascada para describir la presentación visual de documentos HTML en la web

D

DATA PREPROCESSING Preprocesamiento de datos - Conjunto de técnicas para limpiar, transformar y preparar datos antes del entrenamiento de modelos de aprendizaje automático.

DATASET Conjunto de datos estructurado y etiquetado utilizado para entrenar, validar o evaluar modelos de aprendizaje automático.

DIAGNÓSTICO DIFERENCIAL Proceso clínico sistemático para distinguir entre dos o más enfermedades que presentan síntomas similares o superpuestos.

E

EFECTO ICEBERG Fenómeno clínico donde la información visible del paciente representa solo una pequeña parte del problema médico completo.

EMBEDDINGS Representaciones vectoriales numéricas que capturan el significado semántico de palabras o textos en un espacio multidimensional.

ENCODER-DECODER Arquitectura neural que codifica una entrada en una representación interna y la decodifica en una salida específica para la tarea.

EPOCHS Épocas - Número de veces que el algoritmo de entrenamiento recorre completamente todo el conjunto de datos durante el proceso de aprendizaje.

F

F1-SCORE Métrica que combina precisión (precision) y exhaustividad (recall) para proporcionar una evaluación equilibrada del rendimiento del modelo.

FAISS Facebook AI Similarity Search - Librería optimizada para realizar búsquedas eficientes de similitud en conjuntos masivos de vectores.

FINE-TUNING Ajuste fino - Proceso de reentrenamiento de un modelo preentrenado con datos específicos del dominio para especializarlo en tareas particulares.

G

GPT-2 Generative Pre-trained Transformer 2 - Modelo de lenguaje generativo desarrollado por OpenAI, precursor de versiones más avanzadas.

GRADIENT DESCENT Algoritmo de optimización iterativo que minimiza una función de pérdida ajustando gradualmente los parámetros del modelo.

H

HIPAA Ley estadounidense que regula la protección y confidencialidad de la información médica. Establece estándares para el manejo seguro de datos de salud por parte de sistemas tecnológicos en entornos clínicos.

HTML HyperText Markup Language – lenguaje de marcado estándar para estructurar contenido y documentos en la web

HUGGING FACE Plataforma líder que proporciona modelos de lenguaje preentrenados, herramientas de NLP y una comunidad activa de desarrolladores.

HYPERPARAMETERS Hiperparámetros - Configuraciones del modelo que se establecen antes del entrenamiento y controlan el proceso de aprendizaje.

I-J

IA Inteligencia Artificial - Rama de la informática que desarrolla sistemas capaces de realizar tareas que tradicionalmente requieren inteligencia humana.

JSONL JSON Lines - Formato de archivo donde cada línea contiene un objeto JSON válido e independiente, ideal para datasets grandes.

JUICIO CLÍNICO Capacidad profesional para evaluar información médica compleja y tomar decisiones diagnósticas y terapéuticas fundamentadas.

L

LEARNING RATE Tasa de aprendizaje - Hiperparámetro que controla la magnitud de los ajustes realizados en los pesos del modelo durante el entrenamiento.

LLM Large Language Models (Grandes Modelos de Lenguaje) - Redes neuronales profundas entrenadas con enormes cantidades de texto para dominar el lenguaje humano.

M

MAYO CLINIC Institución médica de prestigio mundial, reconocida por su excelencia clínica y utilizada como fuente confiable de información médica.

MINILM Modelo de lenguaje generalista, eficiente y ligero, diseñado para funcionar con recursos computacionales limitados.

MLP Multi-Layer Perceptron (Perceptrón Multicapa) - Red neuronal con múltiples capas completamente conectadas, arquitectura fundamental del deep learning.

O-P

OVERFITTING Sobreajuste - Situación problemática donde un modelo aprende demasiado específicamente los datos de entrenamiento, perdiendo capacidad de generalización.

PANDAS Librería de Python ampliamente utilizada para manipulación, análisis y procesamiento de datos estructurados de forma eficiente.

PERPLEXITY Métrica que cuantifica la incertidumbre de un modelo de lenguaje al predecir la siguiente palabra en una secuencia de texto.

PROMPT ENGINEERING Disciplina que se enfoca en el diseño y optimización de instrucciones (prompts) para maximizar el rendimiento de modelos de lenguaje.

PUBMED Base de datos bibliográfica de literatura biomédica y ciencias de la vida mantenida por la National Library of Medicine.

R

RAG Retrieval-Augmented Generation (Generación Aumentada por Recuperación) - Técnica que combina modelos de lenguaje con sistemas de recuperación de información externa.

RAGAS Retrieval-Augmented Generation Assessment Score - Framework especializado para evaluar objetivamente el rendimiento de sistemas RAG.

RGPD Reglamento europeo que garantiza la protección de los datos personales. Obliga a que cualquier sistema que procese información sensible, como agentes conversacionales en salud, cumpla con principios de transparencia, minimización y consentimiento informado.

RNN Recurrent Neural Networks (Redes Neuronales Recurrentes) - Arquitectura diseñada para procesar secuencias de datos con memoria temporal.

ROUGE Recall-Oriented Understudy for Gisting Evaluation - Conjunto de métricas para evaluar la calidad de resúmenes automáticos.

RXLIST Base de datos en línea especializada que proporciona información detallada sobre medicamentos, dosificaciones y efectos secundarios.

S

SAPBERT Modelo avanzado basado en PubMedBERT, optimizado específicamente para tareas de alineamiento semántico en el dominio médico.

SCRAPY Framework de Python robusto y escalable para realizar web scraping de forma eficiente en proyectos de gran envergadura.

SELENIUM Herramienta de automatización de navegadores web, especialmente útil para extraer datos de sitios web dinámicos e interactivos.

SILHOUETTE SCORE Métrica que evalúa la calidad de agrupamientos (clustering) midiendo tanto la cohesión interna como la separación entre grupos.

T

TEMPERATURE Hiperparámetro que controla el nivel de aleatoriedad en la generación de texto, influyendo en la creatividad versus consistencia de las respuestas.

TINYLLAMA Versión compacta y eficiente de un modelo de lenguaje grande, diseñada para funcionar con recursos computacionales limitados.

TOKENIZACIÓN Proceso fundamental de dividir texto en unidades más pequeñas (tokens) que pueden ser procesadas efectivamente por modelos de lenguaje.

TOP-P SAMPLING Técnica avanzada de muestreo que selecciona palabras de un subconjunto dinámico de tokens más probables, mejorando la calidad del texto generado.

TRANSFORMERS Arquitectura revolucionaria de red neuronal que utiliza mecanismos de atención para procesar secuencias de datos de forma paralela y eficiente.

t-SNE Técnica de reducción de dimensionalidad que proyecta datos de alta dimensión, como embeddings, en un espacio bidimensional, permitiendo visualizar relaciones semánticas entre palabras de forma intuitiva.

W

WEB SCRAPING Raspado web - Técnica automatizada para extraer datos estructurados de páginas web utilizando herramientas y scripts especializados.

WEIGHT DECAY Decaimiento de pesos - Técnica de regularización que previene el sobreajuste añadiendo una penalización a los pesos grandes del modelo.

Introducción

"La inteligencia es la habilidad de adaptarse al cambio."
— Stephen Hawking (2018)

En los últimos años, la inteligencia artificial (IA) se ha reconocido como una poderosa aliada de la inteligencia humana, destacando por su capacidad para procesar, analizar y almacenar grandes volúmenes de datos. Este avance ha sido posible gracias al crecimiento exponencial de la capacidad computacional, el desarrollo de algoritmos de aprendizaje automático y la disponibilidad de grandes conjuntos de datos.

En el sector médico, estas tecnologías están comenzando a transformar la forma en que se toman decisiones clínicas. Los sistemas sanitarios basados en IA pueden analizar miles de historiales clínicos, investigaciones recientes y síntomas complejos para ayudar a los médicos en sus diagnósticos.

Una forma de entender esta evolución es pensar en la medicina como si fuera un sistema de nombres y apellidos. El "nombre" representa la enfermedad general (por ejemplo, cáncer), mientras que el "apellido" define su particularidad: cáncer de mama triple negativo, cáncer de pulmón no microcítico, etc. Es precisamente ese "apellido" el que marca la diferencia en la evolución esperada de la enfermedad y el plan de tratamiento.

El objetivo no es solo identificar patrones comunes que siguen las enfermedades, sino también adaptar el diagnóstico y tratamiento a las particularidades concretas de cada persona: su genética, su entorno, su historial clínico. En este contexto, los modelos basados en inteligencia artificial no solo mejoran la capacidad de resolución médica, sino que permiten diseñar y evaluar las diferentes estrategias sanitarias a fin de obtener una mayor eficacia y resolución en la lucha contra las enfermedades de una manera mucho más personalizada, ajustadas a la realidad única de cada paciente.

De esta manera, la inteligencia artificial no sustituye al juicio clínico, sino que lo amplifica, ayudando a los profesionales a ver más allá del "nombre" de la enfermedad y a actuar con precisión sobre su "apellido". [1]

[1] Todo el código desarrollado para este Trabajo de Fin de Grado se encuentra disponible en el repositorio de GitHub: https://github.com/MonicaFerrerGC/TFG-diagnostic-chatbot-llm

1.1 La Revolución de la IA en su lucha contra las enfermedades

Un Puente entre la Tecnología y la Atención Humana

Un escenario en el que cada síntoma de un paciente activa un sistema inteligente capaz de analizar millones de casos en segundos; donde los médicos cuentan con un asistente que nunca olvida un estudio científico; y donde los diagnósticos se enriquecen con el conocimiento colectivo de la medicina. Esta visión, que hasta hace poco parecía ciencia ficción, está transformándose en una realidad gracias a la inteligencia artificial.

Se proyecta que para el año 2030, cerca del 25 % de la población europea superará los 65 años de edad [8]. Este fenómeno, conocido como el "tsunami plateado", está impulsando una demanda creciente de servicios de salud que desborda la capacidad humana de los sistemas sanitarios actuales [9].

Los profesionales sanitarios se enfrentan a una tensión creciente: cada vez hay más conocimiento disponible, pero menos tiempo para asimilarlo [10]. Solo en *PubMed*, se indexan más de 10.000 artículos biomédicos al año provenientes de distintas instituciones, y se estima que la cantidad total de literatura médica se duplica aproximadamente cada 73 días [11]. Esta sobrecarga informativa dificulta la toma de decisiones clínicas rápidas y precisas.

Por ello, la inteligencia artificial se plantea como un amplificador cognitivo: una herramienta capaz de filtrar el ruido de datos, identificar patrones relevantes y destacar información crítica en tiempo real. Su papel es ayudar a los profesionales a navegar con mayor eficacia por un entorno clínico cada vez más complejo y dinámico.

Un ejemplo reciente y revelador fue la pandemia de *COVID-19*. En ese contexto, la IA no solo ayudó a detectar brotes tempranos, sino que también aceleró el desarrollo de fármacos y vacunas. Plataformas como *Exscientia*, utilizaron modelos de IA para diseñar en tiempo récord inhibidores dirigidos a enzimas clave del virus, como la *proteasa Mpro*, reduciendo significativamente los tiempos tradicionales de investigación [12].

Además, la IA permite avanzar hacia una medicina verdaderamente personalizada. No basta con identificar la enfermedad, el "nombre", sino comprender su "apellido": el subtipo, la mutación genética, el contexto clínico. Cada paciente es único, y su diagnóstico y tratamiento deben adaptarse no solo a la patología, sino también a su *ADN*, su entorno, su historial y sus respuestas individuales. En este sentido, la IA puede trazar estrategias sanitarias más eficientes y diseñar intervenciones ajustadas a las características concretas de cada persona.

1.2 Estructura del documento

Este Trabajo de Fin de Grado se organiza en varias secciones que abordan de forma progresiva los aspectos teóricos, metodológicos, técnicos y experimentales del estudio:

1. Introduce el contexto general del trabajo, destacando el papel emergente de la IA en el ámbito médico, los retos actuales del diagnóstico clínico y la motivación que impulsa esta investigación.

2. Presenta los fundamentos teóricos, explicando los conceptos clave relacionados con los modelos de lenguaje natural a gran escala (LLM), el ajuste fino (fine-tuning) y la estrategia de Generación Aumentada por Recuperación (RAG), así como su relevancia en el entorno clínico.

3. Describe la metodología empleada, incluyendo la construcción de una base de datos clínica, el diseño experimental, los modelos utilizados y los criterios de evaluación definidos para el estudio comparativo.

4. Corresponde al desarrollo, donde se detallan los procesos de implementación técnica, entrenamiento de modelos, integración de sistemas y preparación del entorno de pruebas.

5. Presenta los resultados obtenidos a partir de la evaluación de los tres enfoques propuestos: el modelo base, el modelo ajustado y el sistema con RAG. Se analizan métricas como la precisión diagnóstica, la contextualización de las respuestas y la eficiencia computacional.

6. Recoge las conclusiones del trabajo, destacando las principales aportaciones, limitaciones encontradas y posibles líneas de mejora. También se plantean futuras líneas de investigación orientadas al desarrollo de asistentes conversacionales médicos.

Esta estructura permite abordar de forma ordenada y comprensible los distintos componentes del estudio, facilitando así su lectura.

1.3 Objetivos

"Vivimos en una sociedad profundamente dependiente de la ciencia y la tecnología, en la que casi nadie sabe nada de estos temas."
— Carl Sagan, El mundo y sus demonios (1995)

La inteligencia artificial está transformando la práctica médica, no como sustituto del juicio clínico, sino como una herramienta que refuerza la capacidad de los profesionales sanitarios para analizar datos, integrar conocimientos y tomar decisiones [13].

Por ello, el propósito principal es identificar qué enfoque de inteligencia artificial ofrece un mayor potencial para revolucionar el diagnóstico clínico: ¿modelos genéricos que imitan el razonamiento humano?, ¿sistemas especializados mediante entrenamiento médico?, ¿o arquitecturas que integran el conocimiento global de la medicina en tiempo real?

Esta comparación permitirá evaluar el potencial de cada enfoque como herramienta de apoyo en la toma de decisiones médicas, con el fin de diseñar un agente conversacional (*chatbot*) que integre la solución más eficaz de los tres enfoques descritos.

Para alcanzar este propósito, se han definido los siguientes objetivos específicos (OE):

1. **OE1:** Desarrollar una base de datos clínica estructurada mediante técnicas avanzadas de *web scraping* (2.2.4).
2. **OE2:** Evaluar la comprensión médica de un *modelo LLM* genérico (2.2.3) sin entrenamiento específico, mediante distintos escenarios clínicos.
3. **OE3:** Entrenar modelos con *fine-tuning* (2.2.5) para simular el razonamiento clínico, exponiéndolos gradualmente a casos complejos y ajustando su precisión con retroalimentación.
4. **OE4:** Diseñar un sistema basado en *RAG* (2.2.6) que consulte en tiempo real fuentes como:
   - 4.8 millones de artículos en PubMed [14],
   - 120 000 estudios en ClinicalTrials.gov [15],
   - 85+ guías clínicas internacionales actualizadas [16].
5. **OE5:** Comparar el rendimiento de tres enfoques (modelo base, *fine-tuned* y RAG) [17] en precisión diagnóstica, contexto, eficiencia y fiabilidad.
6. **OE6:** Desarrollar un agente conversacional clínico que:
   - Analice síntomas
   - Cruce datos con el historial médico
   - Genere diagnósticos diferenciales
   - Justifique sugerencias con evidencia reciente
7. **OE7:** Validar el sistema con pruebas funcionales y evaluar sus implicaciones éticas, de seguridad y aplicabilidad clínica, asegurando que complemente el juicio humano.

Contexto

2.1 Contexto clínico

"El buen médico trata la enfermedad; el gran médico trata al paciente que tiene la enfermedad."
— William Osler (1892)

El proceso diagnóstico no sigue un patrón único ni se comporta igual en todos los pacientes. Cada caso plantea un escenario distinto, influido por cómo se presentan los síntomas, el contexto del paciente y la forma en que comunica lo que le ocurre. Por ello, el médico debe ser capaz de interpretar una realidad compleja y muchas veces incompleta.

Este es el día a día de la atención primaria, donde el 68 % de las consultas se basan en síntomas poco concluyentes que podrían ocultar desde una gripe común hasta un cáncer en fase inicial [18]. Todo esto ocurre en un contexto de tiempo extremadamente limitado; en España, un médico de familia dispone de apenas 6 a 10 minutos por paciente, y en muchos casos, incluso menos [19].

2.1.1 La Paradoja del Primer Contacto

La atención primaria es el primer lugar al que acuden los pacientes y donde se resuelven aproximadamente el 80 % de los problemas de salud. Sin embargo, el 20 % restante incluye casos más complejos, donde el diagnóstico se convierte en una auténtica búsqueda de *"la aguja en un pajar"*. [20]

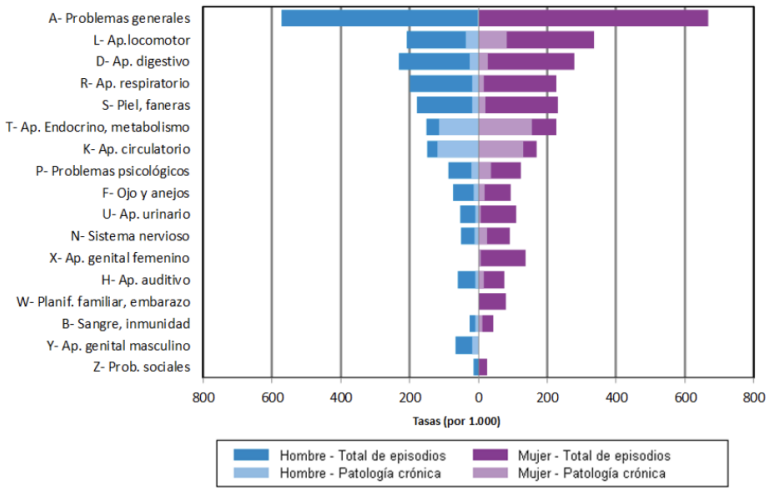


Figura 2.1: Síntomas más comunes en atención primaria [21]

En muchos centros de salud, los médicos de atención primaria atienden entre 30 y 50 pacientes al día, limitando la posibilidad de dedicar tiempo suficiente a casos que requieren una exploración más profunda antes de establecer un diagnóstico adecuado.[22]

El *efecto iceberg* [23] refleja cómo gran parte de la información clínica permanece oculta bajo la superficie (Figura 2.2). En medicina no basta con escuchar lo que el paciente dice, también importa lo que no dice. Al igual que en la serie *House M.D.* [24], "todos mienten": por miedo, vergüenza o desconocimiento, muchos omiten información relevante.

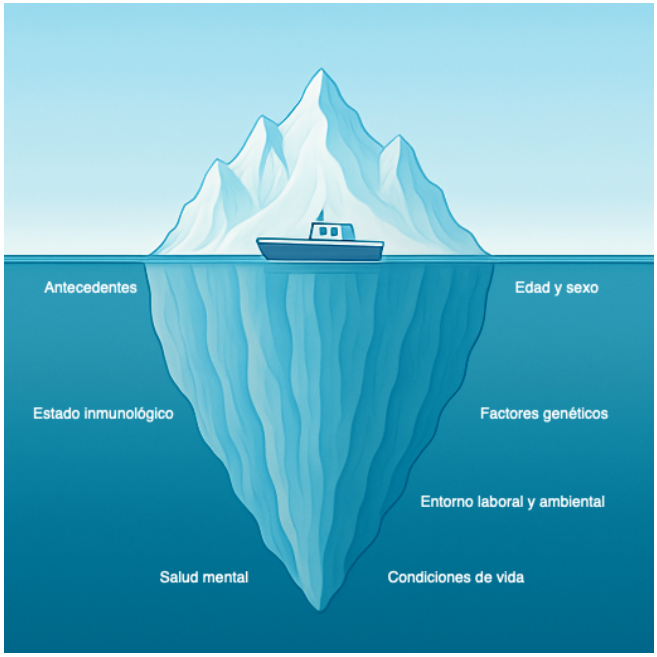


Figura 2.2: Efecto Iceberg: Factores visibles e invisibles que afectan el diagnóstico clínico

2.1.2 Del Síntoma a la Decisión

"El arte del diagnóstico clínico reside en la capacidad de formular las preguntas adecuadas."
— Harriet B. Braiker (2002)

El diagnóstico clínico es un proceso que avanza de forma progresiva, donde cada nuevo síntoma observado aporta una pieza más al "rompecabezas". A medida que el cuadro clínico evoluciona, las hipótesis iniciales se validan, ajustan o descartan.

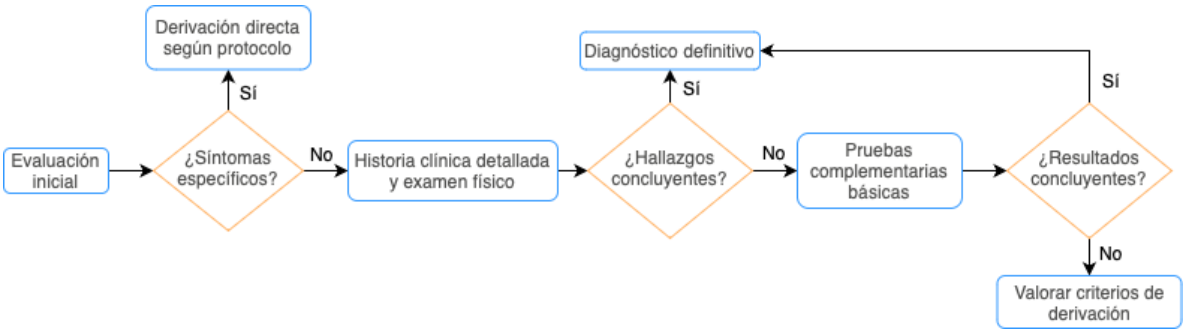


Figura 2.3: Proceso de diagnóstico clínico en atención primaria

2.1.3 El Juicio Clínico en la Incertidumbre

"La medicina es una especialidad de decisiones tomadas con información incompleta, en tiempo limitado y bajo presión de consecuencias."
— Barbara Starfield (2000)

Tomar decisiones médicas no responde a fórmulas ni a soluciones exactas, ya que intervienen múltiples factores: el conocimiento médico, la experiencia acumulada, el contexto personal del paciente y, en muchas ocasiones, una forma de intuición que se cultiva con los años.

Es fundamental que el médico sea capaz de interpretar silencios, contradicciones y matices en el relato del paciente, puesto que la información que el paciente puede omitir, minimizar o incluso distorsionar, ya sea por miedo, vergüenza o desconocimiento, puede llevar a diagnósticos erróneos [25].

La intuición no es una corazonada al azar, surge de la experiencia acumulada tras observar numerosos casos clínicos, de reconocer patrones que no siempre se ajustan perfectamente, y de saber cuándo algo "no encaja del todo", aunque los datos aún no lo confirmen [26].

De esta manera, el médico aprende a convivir con la incertidumbre [27]. No siempre se puede dar un diagnóstico claro en la primera consulta; a veces, lo más prudente es observar y esperar (estrategia *watch-and-wait*) [28].

2.1.4 Diagnóstico Asistido por Inteligencia Artificial

"La inteligencia artificial no reemplazará a los médicos, pero los médicos que la usen reemplazarán a los que no lo hagan."
— Eric Topol (2019)

En un entorno clínico cada vez más complejo, el diagnóstico suele abordarse de forma colaborativa. En muchos hospitales, especialmente ante casos difíciles, se conforman equipos médicos de distintas especialidades que aportan su experiencia, debaten posibilidades y contrastan hipótesis mediante la discusión conjunta [29].

En este contexto, la *inteligencia artificial* no se plantea como un reemplazo del diagnóstico clínico, sino como un miembro más del equipo médico: una herramienta que aporta una perspectiva adicional basada en el análisis de grandes volúmenes de datos [30]. Gracias a su capacidad para procesar historiales médicos, resultados de laboratorio y literatura científica en cuestión de segundos, los sistemas de IA pueden generar hipótesis diagnósticas, sugerir pruebas complementarias y alertar sobre posibles diagnósticos diferenciales que podrían pasar desapercibidos [31].

El diagnóstico asistido por IA representa, por tanto, una evolución del juicio clínico colaborativo. Es una alianza entre conocimiento humano y capacidad computacional, orientada a ofrecer una atención más precisa, personalizada y segura.

2.2 Estado del arte

"La inteligencia artificial no es el futuro, es el presente."
— Stephen Hawking (2017)

A pesar de que la *inteligencia artificial* en la medicina ya forma parte del presente, sigue existiendo aún un gran desconocimiento sobre cómo funciona y cómo puede mejorar la atención sanitaria. ¿Qué significa que una máquina *"aprenda"*? ¿Cómo puede un modelo de lenguaje entender un texto médico? ¿Y por qué es necesario *"ajustarlo"* para que funcione bien en un entorno clínico?

2.2.1 ¿Qué es la Inteligencia Artificial?

La *inteligencia artificial* es una rama de la informática que desarrolla sistemas capaces de realizar tareas que normalmente requieren inteligencia humana, como el aprendizaje, el razonamiento, la percepción o la toma de decisiones. Utiliza algoritmos y modelos matemáticos para analizar datos, reconocer patrones y adaptarse a nuevas situaciones.

La siguiente figura refleja la evolución de la inteligencia artificial a lo largo del tiempo. Esta progresión refleja no solo avances tecnológicos, sino también un cambio en la forma en que concebimos y aplicamos la inteligencia artificial en distintos ámbitos, incluida la medicina.

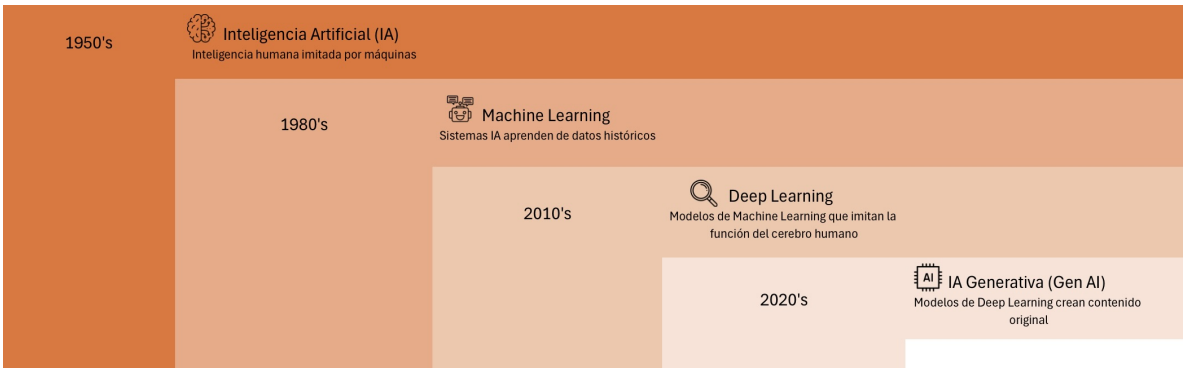


Tabla 2.1: Evolución y relación de la Inteligencia Artificial

2.2.2 Redes Neuronales: Inspiración Biológica, Aplicación Digital

"En lugar de intentar producir un programa que simule la mente del adulto, ¿por qué no empezar con uno que simule la de un niño?"
— Alan Turing (1950)

Las redes neuronales artificiales son modelos matemáticos que imitan, de forma muy simplificada, cómo funciona el cerebro humano [32]. Están formadas por unidades llamadas neuronas artificiales, organizadas en capas. Cada neurona recibe información, la procesa aplicando una operación matemática (llamada función de activación) y luego envía el resultado a otras neuronas en la siguiente capa.

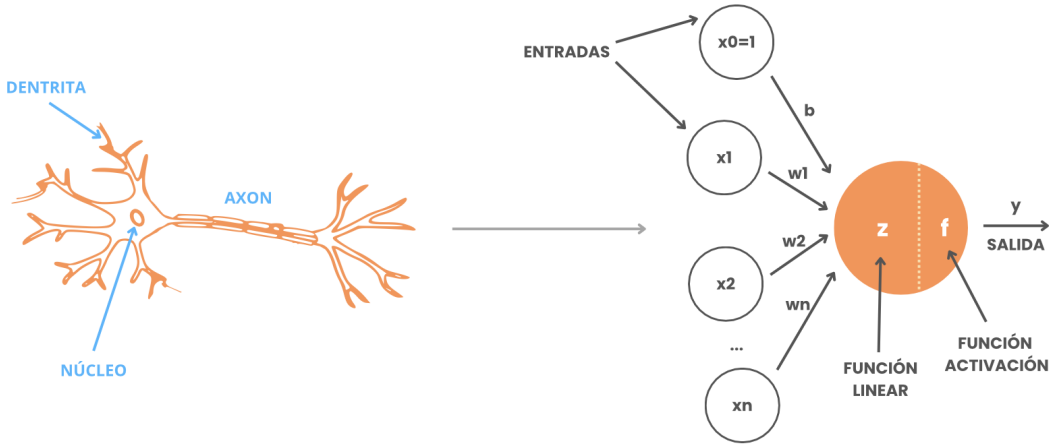


Figura 2.4: Comparación entre una neurona biológica y una neurona artificial

Estas redes aprenden a reconocer patrones complejos en los datos, como si fueran conexiones entre ideas. Por eso, son muy útiles para tareas como clasificar imágenes, predecir valores, identificar patrones o incluso generar texto e imágenes.

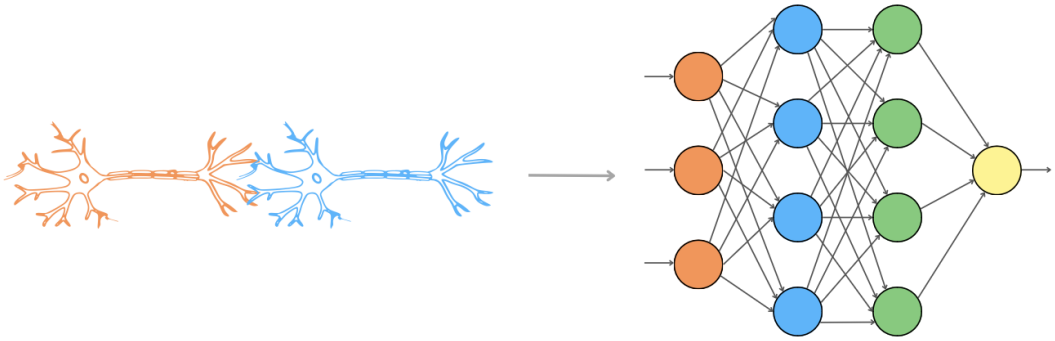


Figura 2.5: Comparación entre las conexiones neuronales biológicas y artificiales

Desde un punto de vista histórico, las redes neuronales han evolucionado a través de distintas etapas clave [33]:

- **Perceptrón (1958)**: es una neurona artificial que combina entradas mediante una suma ponderada y aplica una función de activación. Puede resolver problemas linealmente separables, es decir, aquellos en los que se puede trazar una línea (en 2D) o un plano (en dimensiones mayores) que separe claramente las clases. Sin embargo, no puede aprender patrones no lineales.
- **Perceptrón Multicapa (MLP)**: extiende el perceptrón añadiendo una o más capas ocultas, lo que permite modelar relaciones no lineales complejas. Ajusta sus parámetros internos mediante un proceso iterativo que minimiza el error de predicción, utilizando el cálculo del gradiente para actualizar los pesos. Se emplea en tareas como clasificación, regresión y reconocimiento de patrones.

- **Redes Neuronales Convolucionales (CNN)**: están diseñadas para procesar datos con estructura espacial, como imágenes. Utilizan filtros (o kernels) que detectan características locales, como bordes o texturas. Son ampliamente utilizadas en visión por computadora y reconocimiento de imágenes.
- **Redes Neuronales Recurrentes (RNN)**: están orientadas al procesamiento de datos secuenciales, como texto o series temporales. Incorporan conexiones que permiten conservar información de pasos anteriores. Aunque útiles en tareas como traducción automática o análisis de texto, pueden tener dificultades para aprender dependencias a largo plazo debido a que los gradientes se atenúan progresivamente durante el entrenamiento, lo que limita su capacidad de aprendizaje.
- **Transformadores**: introducen un mecanismo de atención que permite analizar simultáneamente todas las posiciones de una secuencia, sin necesidad de procesarla paso a paso. Son altamente eficientes y han revolucionado el procesamiento del lenguaje natural. Modelos como BERT y GPT se basan en esta arquitectura.

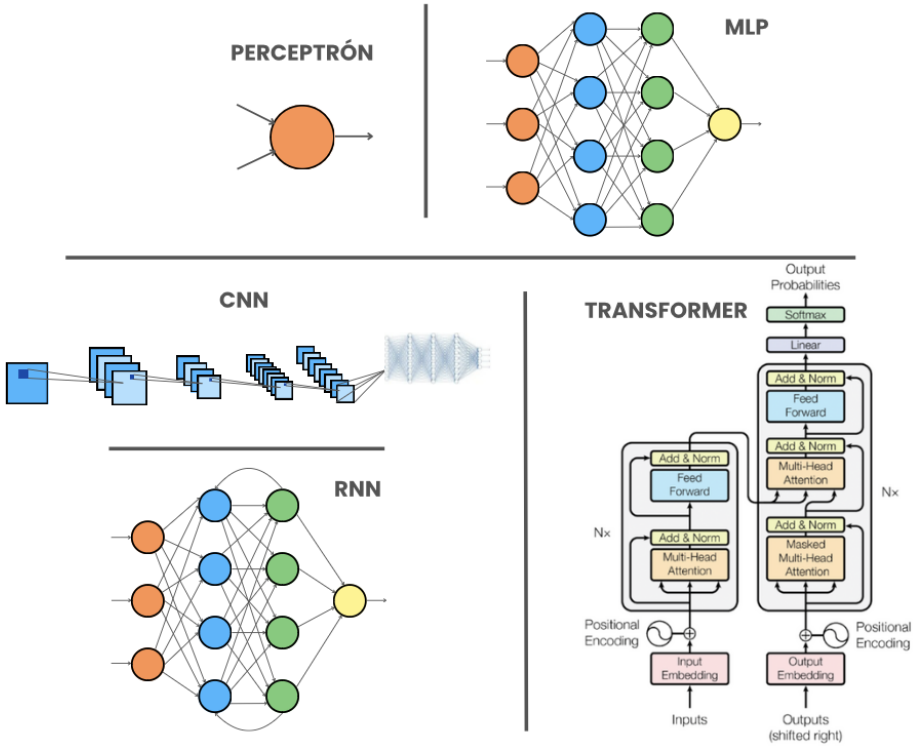


Figura 2.6: Evolución de las arquitecturas de redes neuronales artificiales

Estas arquitecturas representan avances clave en la evolución de las redes neuronales y han permitido abordar una amplia variedad de problemas. Desde tareas simples de clasificación hasta aplicaciones complejas como el reconocimiento de imágenes, la traducción automática o la generación de texto, cada una de estas redes ha contribuido significativamente al avance del aprendizaje automático y al desarrollo de sistemas más precisos, eficientes y flexibles.

2.2.3 Grandes Modelos de Lenguaje (LLM)

Los *grandes modelos de lenguaje (Large Language Models, LLM)* son *redes neuronales* profundas que han sido *entrenadas*, es decir, expuestas a enormes cantidades de texto para que aprendan cómo funciona el lenguaje humano [34].

Este proceso de entrenamiento consiste en mostrar al modelo millones (o incluso billones) de ejemplos de frases, párrafos y documentos, y ajustar sus parámetros internos cada vez que comete un error al predecir la siguiente palabra o interpretar el significado de una oración.

Con el tiempo, el modelo aprende a reconocer patrones lingüísticos, estructuras gramaticales y relaciones semánticas, lo que le permite generar texto coherente y comprender el contexto de lo que se le pregunta [35].

Modelo	Organización	Año	Parámetros (estimado)	Características claves	Uso principal
GPT-4	OpenAI	2023	~1T	Multimodal, potente en razonamiento y comprensión de contexto	Asistente general, codificación, redacción
Mistral 7B	Mistral AI	2023	7B	Modelo denso, rápido y open source	Chatbots ligeros, aplicaciones locales
Gemini 1.5	Google DeepMind	2024	>1T	Contexto extendido (hasta 1M tokens), razonamiento y codificación	IA general, análisis de documentos largos, visión
LLaMA 3	Meta	2024	8B / 70B	Open source, eficiente y de alta calidad	Base para IA personalizada, investigación académica
DeepSeek-VL	DeepSeek AI	2024	7B	Multimodal (imagen + texto)	Visión + lenguaje, análisis de imágenes médicas o técnicas

Tabla 2.2: Comparativa de los principales modelos de lenguaje

A diferencia de los modelos tradicionales, diseñados para tareas específicas como clasificar correos o traducir frases, los LLM son modelos generalistas. Esto significa que pueden adaptarse a distintos tipos de tareas lingüísticas mediante técnicas como el prompting (instrucciones en lenguaje natural 2.2.7) o el fine-tuning (ajuste con datos especializados, como textos médicos 2.2.5).

Estos modelos se basan en la arquitectura *Transformer* (Figura 2.7), la cual emplea mecanismos de atención [36]. Estos mecanismos calculan pesos que indican la importancia relativa de cada palabra en la entrada con respecto a las demás, permitiendo analizar simultáneamente todas las palabras de una secuencia. Esta capacidad permite al modelo considerar el contexto completo de una oración, en lugar de procesarla palabra por palabra de forma secuencial.

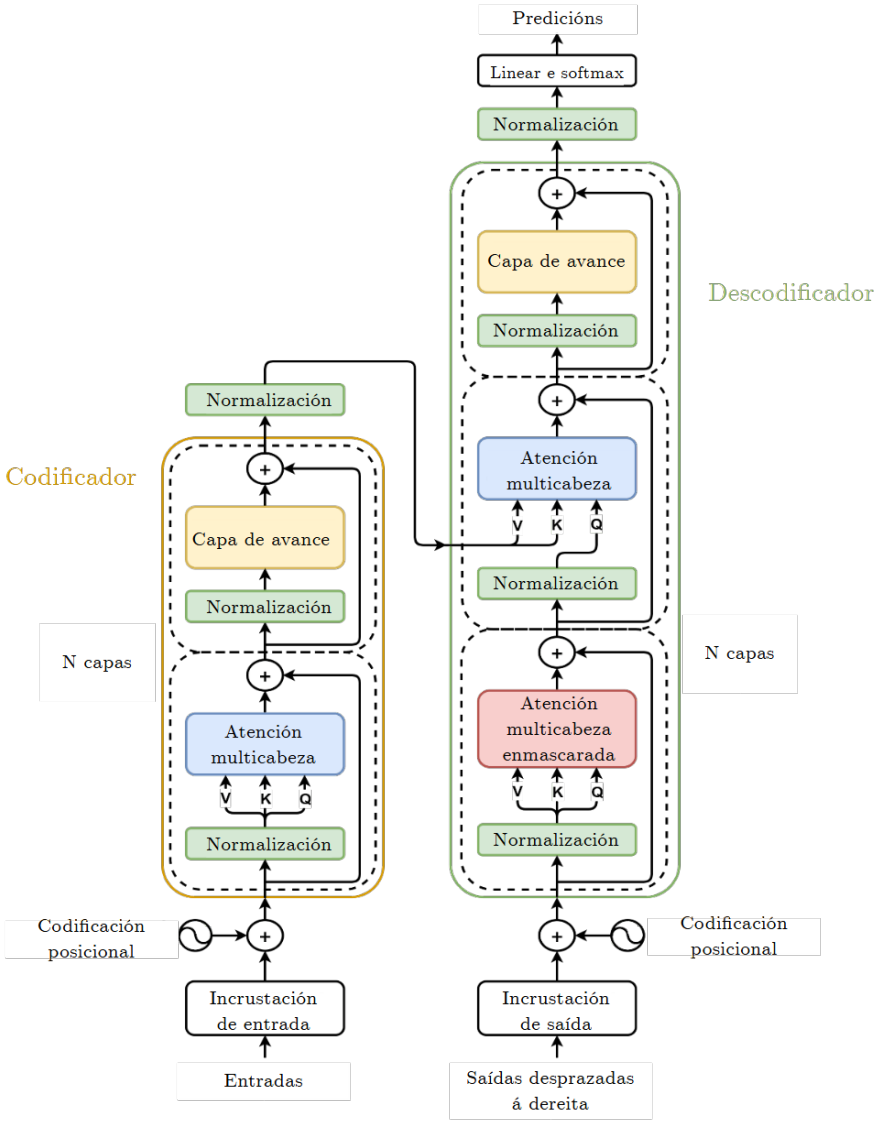


Figura 2.7: Arquitectura del modelo Transformer [1]

Gracias al uso de mecanismos de atención, los *LLM* pueden encontrar relaciones complejas entre palabras, incluso cuando están separadas por muchas otras en una oración. Esto les permite comprender mejor el contexto y generar respuestas más coherentes y precisas, lo que representa un avance significativo en el procesamiento del lenguaje natural.

En el ámbito médico, los modelos son capaces de interpretar textos clínicos complejos (como informes de laboratorio), generar contenido médico contextualizado, extraer información estructurada de historias clínicas y asistir en la toma de decisiones clínicas mediante el análisis de literatura científica. Estos avances no solo optimizan la gestión de la información médica, sino que también abren la puerta a una medicina más personalizada [37].

2.2.4 Generación de Datos Clínicos: Web Scraping Ético y Eficiente

"Los datos son el nuevo petróleo."
— Clive Humby (2006)

La generación de datos clínicos extraídos de fuentes fiables es una tarea fundamental para entrenar y evaluar modelos de lenguaje en el ámbito médico. Una de las técnicas más utilizadas para obtener información textual es el *raspado web (web scraping)*, que consiste en extraer datos de páginas web de forma automatizada mediante fragmentos de código o herramientas especializadas [38].

En términos simples, el *web scraping* permite "leer" el contenido de sitios web (como artículos, tablas o listas) y convertirlo en datos estructurados que pueden servir para análisis o entrenamiento de modelos.

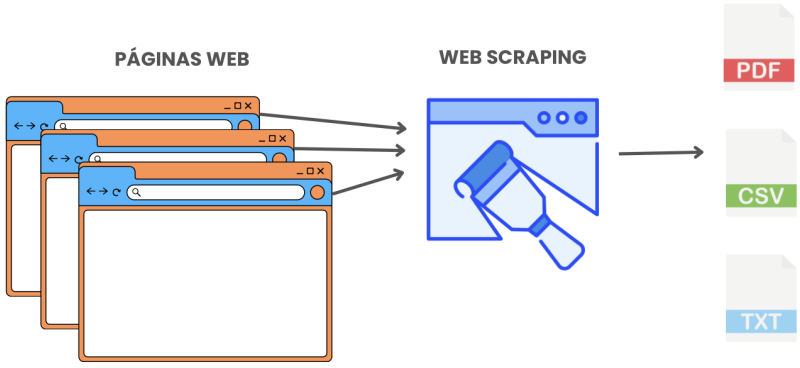


Figura 2.8: Esquema simplificado del proceso de web scraping

El *raspado web* en el ámbito de la salud debe realizarse exclusivamente sobre fuentes fiables, actualizadas y verificadas, como sitios oficiales de organismos de salud (World Health Organization [39], Centers for Disease Control and Prevention [40], Ministerio de Sanidad de España [41]), revistas científicas de acceso abierto (PubMed Central [42], Scientific Electronic Library Online [43]).

Existen múltiples herramientas para realizar *scraping* de forma eficiente. Algunas de las más utilizadas son:

- `BeautifulSoup` y `Scrapy` (Python): para extraer contenido HTML.
- `Selenium`: para interactuar con sitios web que requieren interacción dinámica, por ejemplo hacer clic.
- `Pandas`: para estructurar y limpiar los datos extraídos.

El *web scraping* debe realizarse con responsabilidad ética y dentro del marco legal. Es importante respetar los términos de uso de los sitios, evitar recolectar datos personales y priorizar fuentes abiertas o con licencias compatibles. Cuando se aplica correctamente, permite construir conjuntos de datos clínicos valiosos para entrenar modelos de lenguaje sin comprometer la privacidad ni la integridad de la información [44].

2.2.5 Ajuste fino (Fine-Tuning): Enseñar medicina a un modelo

"El aprendizaje es experiencia. Todo lo demás es información."
— Frase atribuida a Albert Einstein.

El *ajuste fino (fine-tuning)* es el proceso mediante el cual un modelo de lenguaje generalista, capaz de hablar sobre una amplia variedad de temas, se convierte en un especialista. Es como enviar a un estudiante brillante a la facultad de medicina: ya tiene una base sólida, pero ahora necesita aprender los conceptos médicos específicos [45].

Desde una perspectiva técnica, el *fine-tuning* implica reentrenar el modelo base con textos clínicos seleccionados cuidadosamente, como literatura científica, guías médicas y protocolos institucionales. Estos datos actúan como el *"material de estudio"* que permite al modelo interiorizar no solo el vocabulario especializado, sino también los matices, estructuras y formas de razonamiento propias del lenguaje clínico.

Figura 2.9: Esquema general de un LLM generalista

Durante esta fase de aprendizaje, el modelo comienza a reconocer patrones propios de la medicina: cómo se describen los síntomas, cómo se estructuran los diagnósticos diferenciales. Aprende a distinguir entre términos similares pero clínicamente distintos.

El resultado es un modelo que no solo *"sabe de medicina"*, sino que también comprende su contexto, sus implicaciones y sus necesidades.

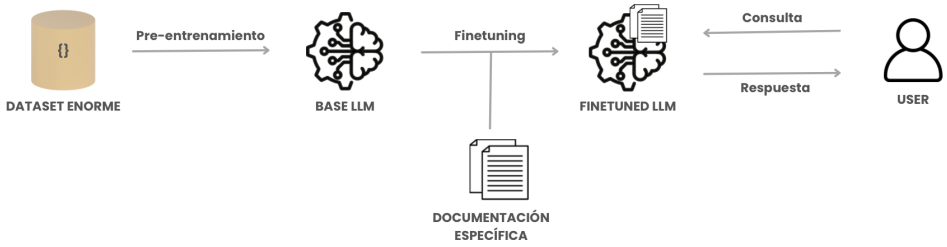


Figura 2.10: Esquema general del proceso de fine-tuning de un modelo LLM

En un entorno donde el conocimiento médico se actualiza constantemente, el *fine-tuning* permite adaptar modelos de lenguaje generalistas a contextos clínicos específicos, dotándolos de un conocimiento especializado que, aunque valioso, requiere actualizaciones periódicas para mantenerse vigente [46].

2.2.6 RAG: Cuando el modelo no lo sabe, lo busca

"El conocimiento tiene un principio, pero no un final."
— Frase atribuida a Geeta Iyengar

Aunque los modelos de lenguaje como los *LLM* pueden generar respuestas coherentes y contextualmente relevantes, tienen una limitación importante: su conocimiento está restringido a los datos con los que fueron entrenados. No pueden acceder a información nueva, específica o actualizada después de su fecha de entrenamiento. Aquí es donde entra en juego una técnica poderosa: la *Recuperación Aumentada por Generación (Retrieval-Augmented Generation, RAG)* [47].

RAG es una técnica que combina *grandes modelos de lenguaje* (sección 2.2.3) con recuperación de información externa. En lugar de confiar únicamente en lo que el modelo *"recuerda"*, *RAG* le permite buscar en fuentes externas, como bases de datos médicas, artículos científicos o documentos clínicos, y utilizar esa información como contexto para generar respuestas más precisas y actualizadas.

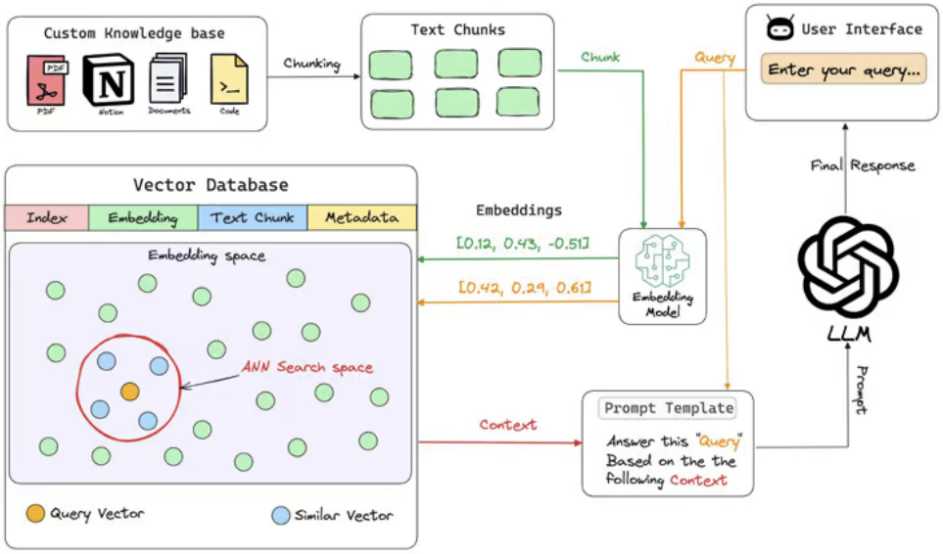


Figura 2.11: Diagrama del flujo de trabajo de un sistema RAG [48]

El proceso *RAG* (Figura 2.11) comienza con la entrada de datos, normalmente en formato `.jsonl`, que contiene información estructurada, como textos médicos o documentos clínicos. Cada fila de esta entrada de datos se transforma en texto limpio y legible.

A continuación, se realiza un análisis del contenido, extrayendo características como la longitud del texto, la presencia de números o los temas principales. Posteriormente, el texto se segmenta en oraciones y se divide en fragmentos más pequeños, conocidos como *chunks*, que facilitan su manejo.

Cada uno de estos fragmentos se convierte en un *embedding*, es decir, una representación

numérica que captura su significado semántico. Estos vectores se almacenan en una base de datos especializada, como *FAISS* [49], que permite búsquedas rápidas por similitud.

Cuando un usuario realiza una consulta, esta también se convierte en un *embedding* y se compara con los almacenados para recuperar los fragmentos más relevantes. Estos fragmentos se agrupan para formar un contexto informativo, que el modelo de lenguaje utiliza para generar una respuesta precisa, fundamentada y adaptada a la pregunta original.

De una manera más simple, este proceso puede expresarse como un modelo con acceso a una biblioteca médica especializada y actualizada; cuando recibe una pregunta, primero busca en esa biblioteca los documentos más relevantes y, a partir de ellos, genera una respuesta informada y contextualizada, combinando su capacidad lingüística con información externa precisa. Este enfoque convierte al modelo en una especie de *"médico con acceso inmediato a las últimas actualizaciones, pronósticos o estrategias del sector sanitario"*.

Este enfoque ofrece múltiples ventajas en el ámbito clínico. Al permitir que el modelo acceda a información en tiempo real, se facilita una actualización continua sin necesidad de reentrenamiento, se reduce la probabilidad de generar respuestas incorrectas o inventadas (conocidas como "alucinaciones") y se optimizan los recursos al evitar procesos de entrenamiento costosos y frecuentes [50].

2.2.7 Prompts: Cómo hablar con una IA

"La calidad de la respuesta depende de la calidad de la pregunta."
— Frase atribuida a Albert Einstein.

Interactuar con un *modelo de lenguaje* va más allá de simplemente hacer preguntas: requiere formular instrucciones claras y bien definidas, conocidas como *prompts*, para guiar su respuesta. La calidad, claridad y estructura de ese mensaje influyen directamente en la precisión y utilidad de la respuesta generada [51].

Un buen *prompt* actúa como una brújula: orienta al modelo hacia el tipo de información que se desea obtener, el formato esperado y el nivel de detalle requerido. Por ejemplo, no es lo mismo preguntar *"¿Qué es la diabetes?"* que *"Resume en tres frases los criterios diagnósticos de la diabetes tipo 2 según la ADA 2024"*. Este segundo es más específico, contextualizado y útil en un entorno clínico.

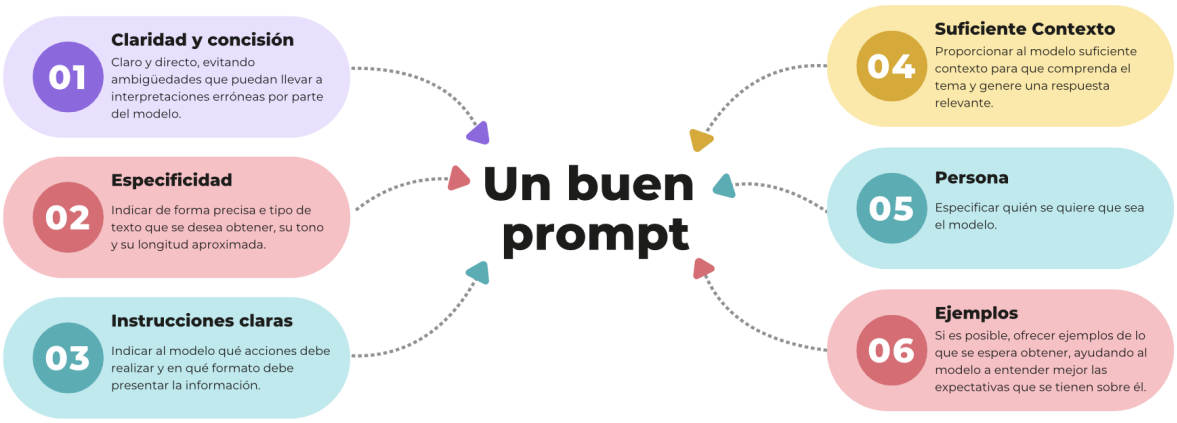


Figura 2.12: Elementos clave para redactar un buen prompt

Como se muestra en la Figura 2.12, un buen *prompt* combina varios elementos clave: claridad, especificidad, instrucciones claras, contexto suficiente, definición del rol del modelo y ejemplos. Estos componentes ayudan a reducir ambigüedades, mejorar la calidad de las respuestas y adaptar la salida del modelo a las necesidades del médico o paciente.

En el contexto médico, donde la precisión y la claridad son esenciales, saber *"cómo preguntar"* puede marcar la diferencia entre una respuesta útil y una potencialmente errónea o ambigüa.

Metodología

3.1 Planificación del proyecto

Las tareas se establecen en un orden temporal y cada una está relacionada con alguno de los objetivos a cumplir. La Tabla 3.1 muestra el catálogo de tareas y los resultados previstos:

OS	TAREA	RESULTADO
OS1	Recopilación y estructuración de datos médicos	Generar una base de datos clínica estructurada mediante web scraping, validada por especialistas y libre de sesgos.
	Preprocesamiento y validación del dataset	Asegurar la integridad, relevancia y utilidad de los datos para el entrenamiento de modelos.
OS2	Evaluación del modelo LLM genérico	Medir la comprensión médica inicial del modelo sin entrenamiento específico, usando escenarios clínicos reales.
OS3	Entrenamiento especializado con fine-tuning	Adaptar el modelo al razonamiento clínico progresivo, priorizando precisión sobre velocidad.
OS4	Implementación del sistema RAG	Desarrollar un sistema híbrido que consulte fuentes médicas globales en tiempo real y reduzca errores por desactualización.
OS5	Comparación de enfoques	Evaluar precisión, eficiencia y fiabilidad de los tres enfoques: modelo base, fine-tuned y RAG.
OS6	Desarrollo del agente conversacional clínico	Crear un chatbot funcional capaz de analizar síntomas, generar diagnósticos diferenciales y justificar sus respuestas con evidencia.
OS7	Validación funcional y análisis ético-clínico	Realizar pruebas con expertos y elaborar un informe sobre seguridad, ética y aplicabilidad clínica del sistema.

Tabla 3.1: Listado cronológico de tareas del proyecto

3.2 Materiales empleados

Entorno de desarrollo	
Entorno principal	JupyterLab
Ventajas	Integración de código, visualización y documentación; entorno interactivo y ágil
Usos	Procesamiento, entrenamiento y evaluación de modelos LLM
Recursos computacionales	
GPU	NVIDIA A100-PCIE-40GB
Memoria GPU total	40 GB
Memoria GPU asignada	10.9 GB
Driver NVIDIA	535.161.07
Versión CUDA	12.2
Ventajas	Reducción significativa en tiempos de entrenamiento y ejecución eficiente de RAG
Librerías y herramientas	
Transformers (Hugging Face)	Carga, entrenamiento y evaluación de modelos LLM
Pandas y NumPy	Procesamiento y análisis de datos
Matplotlib y Seaborn	Visualización de resultados
FAISS	Implementación del sistema de recuperación semántica en el enfoque RAG
Ventajas	Flujo reproducible, escalable y alineado con buenas prácticas en IA clínica

Tabla 3.2: Listado de materiales empleados durante el proyecto

Desarrollo

"La inteligencia artificial es la última frontera. Más allá de ella se abren posibilidades inmensas para mejorar nuestras vidas, pero también desafíos que debemos afrontar con responsabilidad."
— Stephen Hawking (2018)

El presente capítulo describe el proceso de desarrollo llevado a cabo para investigar y comparar los tres enfoques distintos de procesamiento del lenguaje natural aplicados al apoyo en el diagnóstico clínico (modelo genérico, *fine-tune* y *RAG*).

4.1 Arquitectura General del Sistema

La arquitectura del sistema propuesto se ha diseñado con el objetivo de asistir en el diagnóstico clínico. Esta arquitectura permite comparar el rendimiento de modelos con diferentes niveles de especialización y capacidades de recuperación de información, garantizando una evaluación rigurosa y flexible.

4.1.1 Componentes Principales del Sistema

El sistema se compone de varios módulos que trabajan de forma coordinada para ofrecer una experiencia conversacional fluida y clínicamente relevante. La Figura 4.1 muestra una visión general de estos componentes.

Componentes del sistema	
Interfaz de Usuario (UI)	Permite al usuario interactuar con el sistema mediante lenguaje natural.
Procesamiento NLP	Interpreta la entrada del usuario y extrae entidades clínicas relevantes.
Motor de Inferencia	Modelo base (modelo generalista)
	Modelo fine-tuned (modelo ajustado con un corpus médico especializado)
	Sistema RAG (arquitectura híbrida, combina generación con recuperación de información externa)
Base de Datos Clínica	Contiene información médica estructurada para entrenamiento y recuperación.
Recuperación Semántica (RAG)	Busca fragmentos relevantes en la base de conocimiento para enriquecer las respuestas.
Sistema de Evaluación	Mide el rendimiento del sistema en precisión, contexto y eficiencia.

Tabla 4.1: Componentes principales del sistema conversacional para diagnóstico clínico.

4.1.2 Requisitos del Sistema

El diseño del sistema parte de una serie de requisitos que garantizan tanto su funcionalidad como su viabilidad en un entorno clínico real. La Figura 4.2 resume los principales criterios funcionales y no funcionales que han guiado el desarrollo del agente conversacional diseñado para apoyar el diagnóstico clínico.

Requisitos del sistema	
Requisitos Funcionales	Interpretación de síntomas descritos en lenguaje natural.
	Generación de diagnósticos diferenciales.
	Justificación de las respuestas mediante fuentes médicas verificadas.
	Consulta de información externa en tiempo real (en el caso del sistema RAG).
Requisitos No Funcionales	Tiempo de respuesta inferior a 2 segundos en entorno local.
	Modularidad para permitir la sustitución de componentes (modelo, base de datos, interfaz).
	Trazabilidad de las respuestas generadas.
	Cumplimiento ético en el tratamiento de datos clínicos.

Tabla 4.2: Requisitos funcionales y no funcionales del sistema conversacional para diagnóstico clínico.

4.2 Creación y Preprocesamiento del Dataset

"El objetivo es convertir los datos en información y la información en conocimiento."
— Carly Fiorina (2004)

La calidad y la estructura del conjunto de datos son uno de los pilares fundamentales para el *entrenamiento de modelos de lenguaje natural (LLM)*. Por ello, en este proyecto se ha implementado un flujo de trabajo completo de recolección, limpieza, estructuración y enriquecimiento de datos clínicos, lo que ha permitido construir un *dataset* robusto y representativo para el entrenamiento de asistentes conversacionales en el ámbito médico [52].

Figura 4.1: Fases principales para el desarrollo del dataset

4.2.1 Obtención de Datos mediante Web Scraping

En este proyecto, se empleó la técnica de *web-scraping* (2.2.4) para recolectar información clínica detallada desde dos fuentes fiables y ampliamente utilizadas en el ámbito médico: Mayo Clinic [53] y Rx List [54].

Se generan dos archivos CSV fundamentales (Figura 4.3):

1. *diseases_full.csv*: Contiene información obtenida de Mayo Clinic acerca de 1.150 enfermedades.

2. *drugs.csv*: Resultado del *scraping* de la web RxList, recoge un total de 3.929 fármacos.

name	overview	symptoms	causes	risk_factors	complications	prevention	diagnosis	treatment	self_care	others

Generic Name	Brand Name	Drug Class	Description	Uses	Dosage	Side Effects	Interactions	Warnings	Precautions

Tabla 4.3: Vista parcial de los archivos *diseases_full.csv* y *drugs_full.csv*.

Para extraer información de Mayo Clinic y RxList, se desarrollaron herramientas de *scraping* adaptables a cambios en la estructura *HTML*. Se emplearon selectores que combinan etiquetas genéricas como `<h2>` y `<section>` con clases *CSS* específicas, lo que permitió identificar con precisión los bloques de contenido relevantes.

Para construir los archivos, se desarrollaron dos *scripts* en Python, usando las librerías `requests` [55], `BeautifulSoup` [56] y `csv` [57], basándose en una lógica ordenada de navegación y extracción web, que permite realizar una limpieza eficiente de los datos.

Cabe señalar que, durante esta fase, se implementaron sesiones persistentes con reintentos automáticos y pausas aleatorias entre solicitudes, con el fin de evitar bloqueos por parte de los servidores.

4.2.1.1 Extracción de Enfermedades desde Mayo Clinic

Para recopilar información sobre enfermedades, se accedió a la página de índice de Mayo Clinic y se recorrieron todas las letras del alfabeto para obtener los enlaces individuales hacia las fichas clínicas.

En cada una de estas fichas se identificaron dos secciones clave:

- **Symptoms & causes**: Este apartado describe los síntomas clínicos de la enfermedad, su evolución y los criterios que indican cuándo acudir al médico. También aborda las causas posibles —como predisposición genética, agentes infecciosos o ambientales— y los principales factores de riesgo.
- **Diagnosis & treatment**: Incluye las estrategias diagnósticas empleadas, como pruebas clínicas, de laboratorio e imagen, así como los criterios diferenciales. En cuanto al tratamiento, se presentan las opciones terapéuticas disponibles —farmacológicas, quirúrgicas o conductuales—, junto con recomendaciones de manejo y seguimiento clínico.

De las cuales se extrajeron subsecciones como *overview*, *symptoms*, *causes*, *risk_factors*, *complications*, *diagnosis*, *treatment* y *self-care*.

4.2.1.2 Extracción de Medicamentos desde RxList

Para la recolección de información farmacológica, se accedió a la sección de medicamentos de RxList, iterando por cada letra del alfabeto para capturar enlaces a las fichas individuales de los 3.929 fármacos distintos.

Una vez dentro de cada ficha, se extrajeron datos estructurados de distintos apartados, incluyendo *Generic Name*, *Brand Name*, *Drug Class*, *Description*, *Uses*, *Dosage*, *Side Effects*, *Interactions*, *Warnings* y *Precautions*.

4.2.2 Limpieza de Datos

Una vez recolectados los datos, se procedió a realizar un primer filtrado para asegurar su calidad y coherencia. Generando así dos nuevos ficheros (*diseases_ cleaned.csv* y *drugs_ cleaned.csv*).

Las tareas de limpieza incluyen:

- **Eliminación de duplicados:** Se eliminan filas repetidas basándose en las columnas clave (name para enfermedades y Generic Name para fármacos).
- **Supresión de textos irrelevantes o ruidos publicitarios**, como fragmentos de contenido genérico no relacionado, por ejemplo: *"Your gift can go 3x as..."*.
- **Relleno de valores faltantes** con *"no information provided"*: Esta decisión está motivada por la necesidad de evitar celdas vacías en un dataset para modelos LLM. Establecer explícitamente que la información no está disponible mejora la estabilidad y calidad del modelo, al proporcionar un input consistente que permite al modelo aprender a gestionar la incertidumbre o la falta de datos explícitos.

4.2.3 Unión y Enriquecimiento del Dataset

Esta etapa es clave para establecer relaciones significativas entre conceptos médicos, enfermedades y tratamientos, a partir de los textos previamente extraídos y normalizados.

4.2.3.1 Detección de menciones cruzadas

En esta fase inicial se analizan los archivos *diseases_ cleaned.csv* y *drugs_ cleaned.csv* con el objetivo de identificar posibles relaciones entre enfermedades y medicamentos.

Para ello, se emplea una estrategia eficiente basada en la detección de coincidencias textuales dentro de los campos descriptivos, utilizando la biblioteca `FlashText` [58]. Esta herramienta permite buscar palabras clave de forma mucho más rápida que métodos tradicionales como expresiones regulares o búsquedas mediante str.contains().

Con el fin de mejorar la precisión del análisis, los nombres de enfermedades y medicamentos se normalizan mediante la descomposición Unicode (NFKD) - que separa caracteres compuestos (por ejemplo, convierte "á" en "a" + tilde)-, la eliminación de diacríticos, y la conversión a minúsculas.

Esto asegura una mayor robustez frente a variaciones ortográficas o de acentuación.

A continuación, se crean dos instancias independientes de `KeywordProcessor` (extracción de palabras clave): una configurada con el vocabulario de enfermedades y otra con el de medicamentos. Estas instancias se utilizan para analizar todas las columnas de texto en ambos conjuntos de datos, buscando menciones relevantes.

Por ejemplo, si un campo descriptivo de un medicamento contiene la frase *"used to treat hypertension"* y el término *"hypertension"* forma parte del vocabulario de enfermedades, se registra automáticamente como una relación implícita.

Como resultado del proceso, se generan dos nuevos archivos:

- *diseases_with_drugs.csv*, donde cada entrada de enfermedad incluye una o más menciones a fármacos.
- *drugs_with_diseases.csv*, donde cada entrada de fármaco incluye una o más menciones a enfermedades.

Estos archivos permiten construir una primera red semántica de relaciones clínicas relevantes.

4.2.3.2 Normalización de relaciones semánticas

Una vez identificadas las menciones cruzadas entre enfermedades y medicamentos, esta etapa se centra en estructurar dichas relaciones de forma uniforme.

Se definen dos conjuntos de columnas:

- En el caso de enfermedades, se agregan todas las columnas con prefijo drugs_found, que contienen los nombres de medicamentos mencionados en las distintas secciones clínicas.
- Para los fármacos, se seleccionan las columnas con prefijo diseases_found, que indican enfermedades detectadas en campos como Uses, Side Effects o Description.

La función *clean_and_merge* se encarga de procesar fila por fila las columnas seleccionadas, extrayendo y combinando los elementos presentes en listas, eliminando duplicados y descartando valores vacíos.

Como resultado, se genera una única columna por conjunto de datos: *relation_drugs* en el dataset de enfermedades y *relation_diseases* en el dataset de fármacos.

Estas nuevas columnas contienen las menciones cruzadas consolidadas, separadas por el delimitador " | ", lo que permite una exploración ágil sin perder la trazabilidad de las múltiples relaciones posibles.

Además, para mantener la coherencia estructural del dataset, especialmente importante para modelos LLM, se reemplazan los valores nulos, listas vacías o cadenas vacías con el marcador *"no relations obtained"*. Esta normalización evita errores en etapas posteriores de análisis o entrenamiento.

Finalmente, se eliminan las columnas originales utilizadas para generar las relaciones, dejando un esquema de datos más compacto y semánticamente claro.

name	overview	symptoms	causes	risk_factors	complications	prevention	diagnosis	treatment	self_care	others	relation_drugs

Generic Name	Brand Name	Drug Class	Description	Uses	Dosage	Side Effects	Interactions	Warnings	Precautions	relation_diseases

Tabla 4.4: Vista parcial de los archivos *clean_diseases_with_relation.csv* y *clean_drugs_with_relation.csv*

Los nuevos archivos (Figura 4.4) sirven como punto de partida para la etapa de fusión contextual entre ambos dominios, proporcionando una representación limpia y normalizada de las relaciones identificadas.

4.2.3.3 Vinculación contextual

El objetivo de esta etapa es construir un dataset relacional que combine información textual sobre enfermedades y medicamentos, estableciendo vínculos bidireccionales entre ambos.

Para lograrlo, se aplican dos enfoques complementarios:

- **Centrado en enfermedades:** cada enfermedad se toma como entidad principal, asociándola con los medicamentos relacionados.
- **Centrado en medicamentos:** cada fármaco es la entidad principal, vinculándose con las enfermedades asociadas.

Sin embargo, antes de vincular datos, se normalizan los campos clave para asegurar coincidencias exactas:

- En *diseases_cleaned.csv*, se genera *name_norm* a partir del campo *name*, aplicando el mismo criterio.
- En *drugs_cleaned.csv*, se crea la columna *generic_norm* a partir del nombre genérico del medicamento, en minúsculas y sin espacios sobrantes.

Esta normalización evita errores causados por diferencias ortográficas o de formato.

A continuación, se describen las dos fases de fusión contextual aplicadas:

1. **Fusión basada en enfermedades**:

   Usando *clean_diseases_with_relation.csv*, donde cada enfermedad tiene una columna *relation_drugs* con los fármacos mencionados:

   - Se convierte *relation_drugs* en una lista.
   - Cada nombre se normaliza y compara con *generic_norm* en el dataset de medicamentos.
   - Si hay coincidencia, se crea una fila combinando datos de enfermedad y medicamento. Si no hay coincidencia, se conserva la fila con la información de la enfermedad y las columnas del fármaco vacías.
   - Incluir también las enfermedades sin menciones (o con *no relations obtained*) sin datos farmacológicos.
   - Finalmente, añadir los medicamentos no mencionados por ninguna enfermedad.

2. **Fusión basada en medicamentos**:

   En este caso, se utiliza el archivo *clean_drugs_with_relation.csv*, con la columna *relation_diseases* por medicamento:

   - Se transforma esa columna en una lista de enfermedades.
   - Cada una se normaliza y se compara con *name_norm* del dataset de enfermedades.
   - Si hay coincidencia, se combinan los datos del medicamento y la enfermedad. Si no, se genera una fila con el fármaco y columnas vacías en la parte clínica.

- Incluir también los fármacos sin relaciones explícitas, vinculándolos a filas vacías en el lado de enfermedades.
- Finalmente, agregar las enfermedades no vinculadas con ningún medicamento.

Como resultado, se generan dos archivos `CSV` que reflejan las relaciones entre ambos dominios:

- *diseases-drugs_ relation.csv*
- *drugs-diseases_ relation.csv*

Estos archivos proporcionan una visión estructurada y enriquecida de las relaciones clínicas, útil para tareas como generación de recomendaciones, inferencia de tratamientos y análisis de efectos adversos. Además, sirven como base para construir el dataset final destinado al ajuste fino de modelos de lenguaje.

4.2.3.4 Depuración estructural y preparación final

Una vez realizada la asociación bidireccional entre enfermedades y fármacos, se deben preparar los archivos, garantizando su limpieza y formato adecuado para tareas de procesamiento semántico o aprendizaje automático.

El proceso se organiza en tres etapas fundamentales:

1. **Eliminación de columnas redundantes:** Se eliminan las columnas *relation_ drugs* y *relation_ diseases*, ya que la relación entre enfermedades y fármacos ya está representada en ambos archivos, por lo que estas columnas resultan innecesarias.
2. **Relleno de valores nulos:** Las celdas vacías se completan con el texto *"no relations obtained"*, asegurando así una interpretación coherente por parte de los modelos.
3. **Exportación de archivos finales:** Los conjuntos de datos depurados se almacenan como:
   - *clean_ union_ diseases-drugs.csv*
   - *clean_ union_ drugs-diseases.csv*

4.2.3.5 Generación del conjunto de datos final

Como paso final, se debe construir un conjunto de datos que combine descripciones de enfermedades con información farmacológica, evitando duplicados y relaciones irrelevantes. El proceso se desarrolla en seis etapas:

1. **Carga de los archivos limpios:** Se importan los archivos *clean_ union_ drugs-diseases.csv* y *clean_ union_ diseases-drugs.csv*, que ya contienen las relaciones entre enfermedades y fármacos previamente unificadas y limpiadas.
2. **Unificación de estructura:** Se asegura que ambos archivos tengan el mismo orden y número de columnas. Si falta alguna columna, se añade con valores vacíos (*None*) para poder combinarlos sin errores.

3. **Fusión y eliminación de duplicados:** Los dos conjuntos se unen en un solo *DataFrame*, y se eliminan las filas duplicadas para que cada relación enfermedad-fármaco aparezca solo una vez.

4. **Eliminación de relaciones sin valor:** Se eliminan los registros donde una enfermedad o fármaco solo tiene la etiqueta *"no relations obtained"*, siempre que esa entidad sí tenga relaciones válidas en otros registros. Esto evita mantener datos que no aportan información útil.

5. **Filtrado de datos irrelevantes:** Se eliminan combinaciones poco realistas o sin sentido clínico, asegurando que el conjunto de datos conserve solo relaciones significativas.

6. **Exportación del dataset final:** El conjunto final, ya limpio y sin redundancias, se guarda como *full_ dataset.csv*. Este archivo está listo para ser convertido a otros formatos (como `JSONL`), analizado o usado para entrenar modelos de lenguaje.

name	overview	symptoms	causes	risk_factors	complications	prevention	diagnosis	treatment	self_care	others	Generic Name	Brand Name	Drug Class	Description	Uses	Dosage	Side Effects	Interactions	Warnings	Precautions

Tabla 4.5: Vista general del archivo *full_ dataset.csv*

El archivo *full_ dataset.csv* (Figura 4.5) representa una recopilación depurada y estructurada de relaciones entre enfermedades y fármacos, lista para ser utilizada en tareas como la extracción de conocimiento, la evaluación de patrones clínicos y el entrenamiento de modelos de lenguaje especializados en el ámbito médico.

4.2.4 Generación del Dataset en Formato JSONL

Como etapa final, se construyó un *dataset* en formato `JSONL`, estructurado específicamente para tareas de *fine-tuning* de modelos de lenguaje (LLMs). Este archivo contiene ejemplos en el formato `instruction → output`, adecuados para tareas supervisadas de generación de texto clínico. Cada entrada representa una interacción informativa derivada del corpus consolidado.

El proceso se organizó en las siguientes fases:

1. **Carga del dataset final:** Se importó el archivo *full_ dataset.csv*, resultado de integrar los datos de enfermedades y medicamentos.

2. **Definición de plantillas:** Se elaboraron múltiples modelos de preguntas con el objetivo de abarcar las principales categorías de información.
   - Enfermedades: síntomas, causas, factores de riesgo, complicaciones, prevención, diagnóstico, tratamiento y autocuidados.
   - Medicamentos: usos, efectos adversos, precauciones, interacciones, advertencias y dosificación.
   - Relaciones enfermedad-fármaco: vinculación directa entre diagnósticos y principios activos utilizados como tratamiento.

3. **Generación de muestras:** Para cada fila del dataset, se selecciona aleatoriamente una plantilla de pregunta correspondiente a una categoría informativa (como síntomas, tratamiento o efectos secundarios), y se genera un ejemplo en formato pregunta-respuesta rellenando los campos de la plantilla con los datos específicos de esa fila (como el nombre de la enfermedad, los síntomas o el nombre del medicamento).

   ```
   {
     "instruction": "What are the symptoms of diabetes?",
     "input": "",
     "output": "Increased thirst, frequent urination, blurred vision..."
   }
   ```

   Este proceso permite crear automáticamente un conjunto de datos variado y coherente, ideal para entrenar modelos de lenguaje en tareas médicas, asegurando diversidad en las preguntas y cubriendo múltiples aspectos relevantes de enfermedades y medicamentos.

4. **División en conjuntos de entrenamiento y prueba:** Se reservó aleatoriamente el 10 % de las muestras para el conjunto de prueba (*test*). En este subconjunto, se utilizaron variantes alternativas de las plantillas para las instrucciones, con el fin de evaluar la capacidad del modelo para generalizar variaciones.

5. **Exportación del archivo:** Cada muestra generada se guarda como una línea independiente en el archivo *full_ dataset.jsonl*, utilizando codificación UTF-8 y formato JSON compatible con caracteres Unicode.

La creación de conjuntos separados de entrenamiento y prueba es fundamental para evaluar de forma objetiva el rendimiento de los modelos. Esta separación garantiza que las métricas obtenidas reflejen la capacidad del modelo para generalizar a nuevas instrucciones, y no simplemente memorizar patrones vistos durante el entrenamiento.

Ambos conjuntos de datos constituyen una base sólida para el desarrollo de asistentes médicos automatizados, sistemas de recomendación sanitaria o herramientas de apoyo al diagnóstico, aprovechando el potencial de los modelos generativos en el ámbito médico.

4.3 Selección y uso del modelo de lenguaje preentrenado

En esta etapa del proyecto, se planteó la necesidad de comparar varios modelos de lenguaje de tipo LLM (*Large Language Model*) que pudieran ser utilizados para la evaluación de su desempeño en tareas específicas del dominio médico.

4.3.1 Modelos evaluados

Se seleccionaron y evaluaron distintos modelos de lenguaje, abarcando tanto modelos de propósito general como especializados en el ámbito médico.

Para este análisis comparativo se consideraron los modelos **BioGPT**, **TinyLLaMA** y **GPT-2**:

Modelo	Dominio Principal	Tamaño (parámetros)	Entrenamiento Inicial	Enfoque/Arquitectura	Casos de Uso Típicos
BioGPT	Biomédico	347M	PubMed abstracts y artículos biomédicos	GPT-2 modificado	Extracción de información médica, QA biomédico
TinyLlama	General	1.1B	The Pile + otros datos públicos	LLaMA-compatible	RAG, chatbot ligero, fine-tuning en dispositivos limitados
GPT-2	General	137M	Web scrape (Books, Wikipedia, etc.)	Transformer Decoder	Tareas de lenguaje general, generación de texto

Tabla 4.6: Comparativa de modelos de lenguaje evaluados

4.3.2 Descarga e implementación

Para llevar a cabo la evaluación de los modelos de lenguaje, se realizó un proceso común de descarga e implementación para todos los modelos seleccionados: **BioGPT**, **TinyLLaMA** y **GPT**.

4.3.2.1 Descarga y almacenamiento local

Todos los modelos seleccionados permiten ejecución local, por lo que fueron descargados desde la plataforma *Hugging Face* utilizando la librería `transformers`. Este proceso ofrece varias ventajas, como:

Figura 4.2: Ventajas de almacenar y ejecutar modelos de lenguaje localmente

4.3.2.2 Carga e interacción con los modelos

Una vez descargados, los modelos fueron cargados en memoria y preparados para su ejecución. Se configuró automáticamente el uso de GPU o CPU en caso que no estuviera disponible.

Para evaluar la capacidad de los modelos en tareas médicas, se diseñaron entradas de prueba con preguntas médicas reales. Estas entradas fueron procesadas mediante un *prompt* (2.2.7) estructurado que guiaba la generación de respuestas, asegurando consistencia en el formato y en la calidad de la información generada.

4.3.2.3 Evaluación automática de resultados

Para evaluar el rendimiento de los modelos de lenguaje, se utilizó un conjunto de datos de prueba; cada entrada incluía una instrucción, un contexto opcional y una respuesta esperada, simulando interacciones reales en el ámbito médico.

El objetivo de esta evaluación fue medir la capacidad de los modelos para generar respuestas coherentes, relevantes y precisas. Para ello, se aplicaron diversas métricas automáticas que permiten comparar las respuestas generadas con las respuestas de referencia (Sección 4.6), con el fin de obtener una valoración objetiva del comportamiento de cada modelo ante distintos tipos de preguntas clínicas (resultados en la sección 5.1).

4.4 Proceso de Fine-Tuning del Modelo de Lenguaje

El *fine-tuning* (visto en la sección 2.2.5) es una etapa clave en la adaptación de modelos de lenguaje preentrenados a tareas específicas.

Mejorando así la capacidad del modelo para generar respuestas médicas coherentes, precisas y alineadas, el *fine-tuning* busca incorporar directamente el conocimiento aprendido, ajustando sus parámetros según esa información.

4.4.1 Preparación del conjunto de entrenamiento

Para el entrenamiento del modelo, se utilizó el conjunto de datos médicos previamente descrito en la sección 4.2. Este dataset fue construido mediante distintas etapas de *scraping*, limpieza y enriquecimiento semántico, a partir de información estructurada obtenida de fuentes fiables como `Mayo Clinic` [53] y `RxList` [54].

El conjunto de datos sigue la estructura `(instruction, input, output)`. En cada instancia, se plantea una cuestión médica (`instruction`), se proporciona contexto adicional si es necesario (`input`) y se genera una respuesta precisa y contextualizada (`output`), basada en el contenido médico previamente aprendido.

A continuación, se presentan algunos ejemplos del formato utilizado:

```
{
  "instruction": "Indica los tratamientos habituales para la hipertensión.",
  "input": "",
  "output": "La hipertensión puede tratarse con fármacos como los inhibidores ..."
}
```

```
{
  "instruction": "Describe los efectos secundarios del fármaco amlodipina.",
  "input": "",
  "output": "Los efectos secundarios más comunes de la amlodipina incluyen ..."
}
```

4.4.2 Configuración del proceso de entrenamiento

Se llevó a cabo un proceso de entrenamiento independiente para cada uno de los modelos base: BioGPT, TinyLLaMA y GPT. Todos fueron entrenados con el mismo conjunto de datos, descrito en la Sección 4.2, con el objetivo de garantizar condiciones equivalentes y una comparación justa en la etapa de evaluación (ver Sección 5).

El conjunto de datos se preparó combinando en un único campo de texto la instrucción, el contexto (cuando aplica) y la respuesta esperada. Este texto se tokenizó utilizando el tokenizador correspondiente a cada modelo.

Para llevar a cabo el entrenamiento, se utilizó la clase `Trainer` de la biblioteca `Transformers`, junto con un `DataCollatorForLanguageModeling` configurado para aprendizaje causal `(mlm=False)`. Se aplicaron los mismos hiperparámetros para todos los modelos, los cuales se resumen en la Figura 4.7.

Number Epochs	20
Batch Size	32
Gradient Accumulation Steps	4
Learning Rate	5,00E-05
Weight Decay	0.01
Max Sequence Length	512
Tokenizer Padding Strategy	max_length

Tabla 4.7: Resumen de los hiperparámetros utilizados durante el fine-tune de los modelos.

A continuación se detallan los parámetros empleados:

- **Número de épocas (Epochs)**: Total de veces que el modelo recorre por completo el conjunto de datos (20).
- **Tamaño de lote (Batch Size)**: Cantidad de muestras procesadas en paralelo en cada iteración (32).
- **Acumulación de gradientes**: Número de pasos antes de actualizar los pesos del modelo, útil para simular lotes más grandes (4).
- **Tasa de aprendizaje (Learning Rate)**: Define la magnitud de los ajustes aplicados a los pesos del modelo en cada actualización (5e-5).
- **Penalización por peso (Weight Decay)**: Técnica de regularización para reducir el sobreajuste (0.01).
- **Longitud máxima de secuencia**: Límite máximo de tokens por entrada tras la tokenización (512).
- **Estrategia de padding**: Se utiliza `max_length` para completar las secuencias hasta alcanzar la longitud máxima definida.

Esta configuración común permite evaluar de manera controlada cómo influye el modelo base en los resultados, manteniendo constantes tanto los datos de entrenamiento como los parámetros utilizados.

4.4.2.1 Exportación del modelo y primeras interacciones

Una vez completado el proceso de *fine-tuning*, cada modelo fue exportado junto con su *tokenizador* correspondiente utilizando las funciones `save_pretrained()` de la biblioteca `Transformers`, permitiendo reutilizar los modelos entrenados sin necesidad de repetir el proceso de ajuste.

Para realizar una primera evaluación, se diseñó un *script* que carga el modelo entrenado y genera respuestas a partir de preguntas médicas formuladas. Las preguntas debían seguir la misma estructura utilizada durante el entrenamiento, es decir, como una instrucción precedida por el prefijo `Prompt:` y seguida de la etiqueta `Response:`, que indica el inicio de la respuesta esperada.

Durante la generación de texto, se utilizaron los parámetros `top_p = 0.95` y `temperature = 0.2` para controlar la diversidad y coherencia de las respuestas.

- `top_p` (nucleus sampling): es una técnica que limita la generación del modelo a las palabras más probables, seleccionando solo aquellas cuya suma de probabilidades alcanza un 95 %. Esto permite que las respuestas sean variadas y naturales, sin perder coherencia.
- `temperature`: controla cuánta aleatoriedad hay al elegir las palabras que el modelo genera. Si se usa un valor bajo, como 0.2, el modelo tiende a dar respuestas más predecibles y seguras. Esto es especialmente importante en el ámbito médico, donde es crucial que las respuestas sean precisas y fiables.

Esta configuración busca lograr un equilibrio adecuado entre variedad y rigor clínico, garantizando que las respuestas generadas sean coherentes y precisas.

Los resultados obtenidos con cada modelo pueden consultarse en la Sección 5.2, donde se evalúan tanto su precisión como su validez médica.

4.5 Integración de RAG

Con el objetivo de mejorar la capacidad diagnóstica del agente conversacional y mantenerlo actualizado con los últimos avances médicos, se ha implementado un sistema basado en *Retrieval-Augmented Generation* (RAG).

Este enfoque permite combinar el conocimiento preentrenado del modelo con información actualizada, obtenida tanto de una base de datos local como de fuentes científicas externas.

En una primera etapa, el sistema accede a una base de conocimiento local construida a partir de textos biomédicos. Estos textos se segmentan en fragmentos y se representan mediante *embeddings* semánticos generados con un modelo especializado en lenguaje médico. Esta representación vectorial permite calcular la similitud entre la consulta del usuario y los fragmentos almacenados, facilitando la recuperación de los textos más relevantes (Anexo C).

Dado que la calidad de los *embeddings* influye directamente en el rendimiento del sistema RAG, se llevó a cabo una evaluación comparativa entre distintos modelos de generación semántica. El objetivo fue identificar el modelo que ofreciera representaciones más precisas y útiles para tareas médicas, optimizando así la recuperación de información (ver Sección 5.3).

De forma complementaria, el sistema accede a literatura científica reciente a través de la API Entrez de NCBI, consultando directamente la base de datos PubMed [59]. Se priorizan los

resúmenes y abstracts más relevantes y actuales, que se integran con los fragmentos locales (Anexo C.3).

Todos los textos recuperados se combinan, se filtran para eliminar duplicados y se estructuran en un único *prompt* que se envía al modelo de lenguaje. Este contexto enriquecido permite generar respuestas más precisas, fundamentadas y alineadas con el conocimiento médico vigente.

Una de las principales ventajas de este enfoque es que permite utilizar modelos base sin necesidad de realizar un costoso proceso de *fine-tuning*. Gracias a la recuperación de contexto externo, el sistema puede adaptarse a distintos dominios, reduciendo significativamente los tiempos de desarrollo y los recursos computacionales necesarios.

El proceso de implementación se dividió en dos fases principales:

1. **Indexación del conocimiento** El corpus biomédico se segmentó en fragmentos (*chunks*) de tamaño fijo. A cada uno se le generó un vector semántico mediante un modelo de *embeddings* biomédico. Estos vectores, junto con su texto asociado, se almacenaron en un índice que permite búsquedas por similitud.
2. **Recuperación y generación** Ante una consulta, se genera su *embedding* y se comparan las distancias con el índice. Los fragmentos más relevantes se incorporan al *prompt* del modelo, que genera una respuesta combinando su conocimiento previo con la información recuperada.

Este enfoque no solo mejora la precisión y actualidad de las respuestas, sino que también aporta transparencia, ya que permite identificar los fragmentos utilizados como contexto.

Este enfoque permite al modelo ofrecer respuestas más precisas y actualizadas, especialmente en situaciones donde su conocimiento preentrenado podría ser insuficiente. Además, al identificar y mostrar los fragmentos utilizados como contexto, se aporta transparencia y trazabilidad a las respuestas generadas.

4.6 Evaluación mediante métricas

Para evaluar el rendimiento de los modelos entrenados, se emplearon dos enfoques complementarios: métricas tradicionales de generación de texto y métricas específicas para *RAG*. Esta combinación permite obtener una visión más completa tanto de la calidad lingüística como de la relevancia contextual de las respuestas generadas.

Para esta evaluación se utilizó el conjunto de datos de prueba (*test set*), que representa el 10 % del dataset original y contiene preguntas nuevas que los modelos no han visto durante el entrenamiento (4.2.4). Esto garantiza una evaluación objetiva de su capacidad de generalización.

Además, para calcular la similitud semántica entre las respuestas generadas y las referencias, se empleó el modelo de embeddings seleccionado en la sección 5.3. Este modelo fue elegido

tras una evaluación comparativa entre varias alternativas, seleccionando aquel que ofreció el mejor rendimiento en tareas médicas.

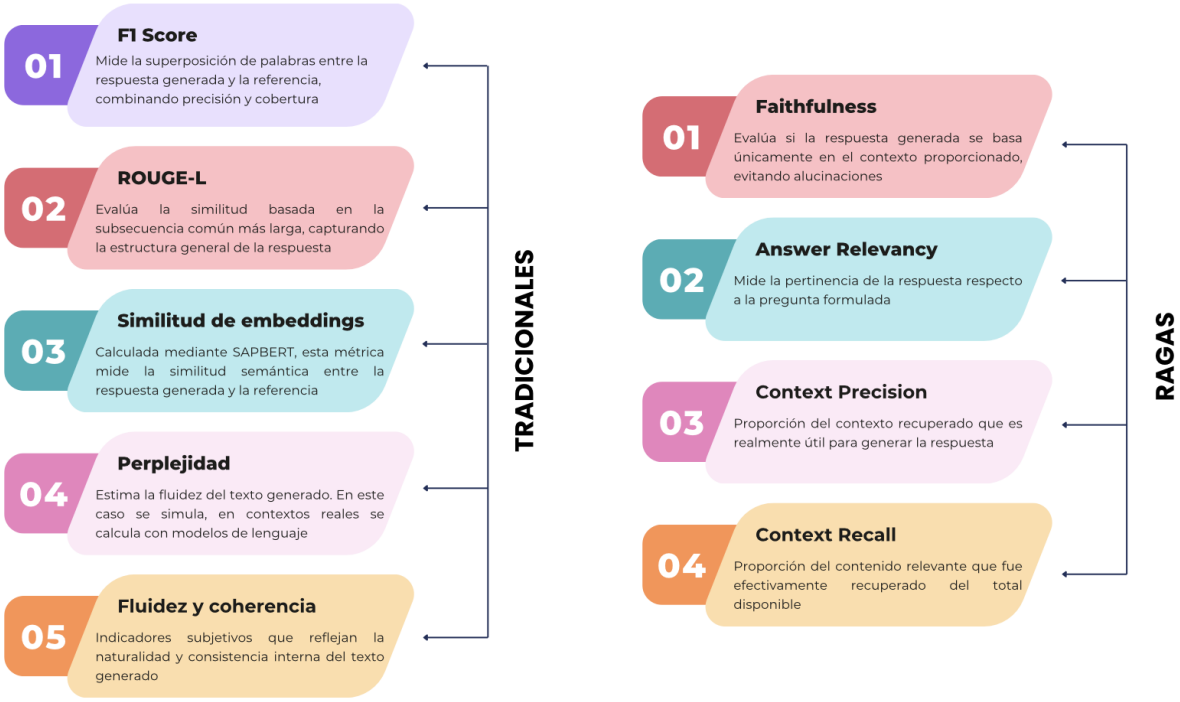


Figura 4.3: Comparación entre métricas tradicionales y métricas RAGAS

Estas métricas permiten analizar no solo la calidad formal de las respuestas, sino también la eficacia del sistema de recuperación de información, aspecto especialmente relevante en entornos médicos donde la trazabilidad, la relevancia y la precisión basada en evidencia científica son fundamentales.

Además de las métricas tradicionales, se utilizó *RAGAS (Retrieval-Augmented Generation Assessment Score)* [60], una herramienta diseñada específicamente para evaluar sistemas *RAG*. Esta biblioteca permite medir de forma objetiva aspectos clave como la relevancia del contenido recuperado, la completitud de las respuestas y su precisión respecto al contexto.

En este proyecto, *RAGAS* se utilizó para evaluar automáticamente las respuestas generadas a partir de consultas médicas, comparándolas con los fragmentos recuperados y referencias médicas. Esto permitió no solo valorar la calidad lingüística, sino también verificar el respaldo documental de cada respuesta, mejorando la trazabilidad y la solidez médica del sistema.

Resultados

Esta sección resume los resultados obtenidos al evaluar tres enfoques para el desarrollo de un agente conversacional clínico: un modelo genérico, uno ajustado mediante fine-tuning con datos médicos, y un sistema basado en RAG. Se analizan sus respuestas ante consultas médicas, así como métricas que permiten comparar su precisión, contextualización y eficiencia.

5.1 Evaluación inicial de modelos base

Para analizar el comportamiento de los modelos de lenguaje seleccionados, se les plantearon dos preguntas médicas sin ningún tipo de prompt estructurado:

1. *Which smallpox vaccine is safer for people with compromised immune systems, and why?*
2. *What are the stages of smallpox rash progression from initial symptoms to scabbing?*

La Figura 5.1 muestra las respuestas generadas por tres modelos: **BioGPT**, **GPT** y **TinyLLaMA**. Esta evaluación permite observar diferencias significativas en cuanto a precisión, coherencia y relevancia clínica.

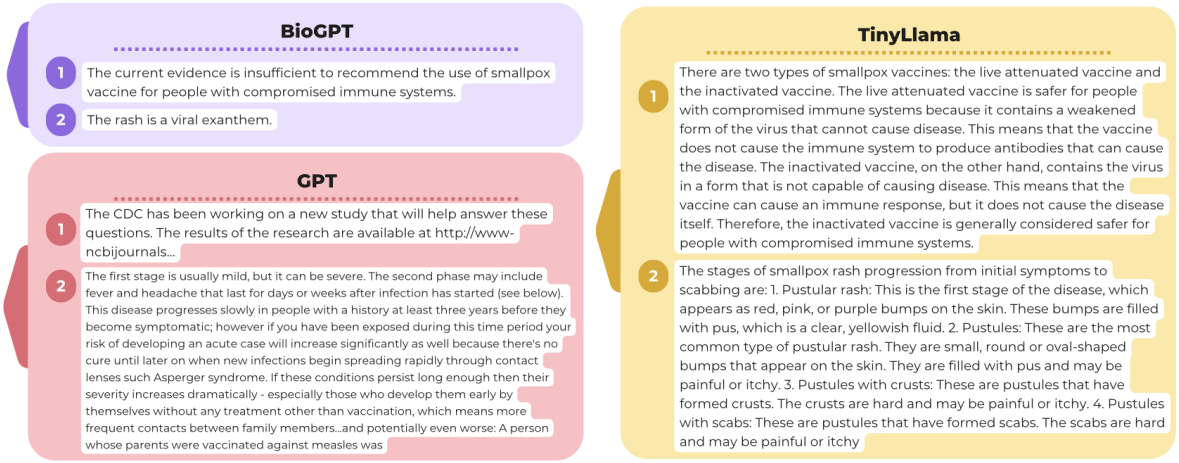


Figura 5.1: Respuestas generadas por modelos base sin prompt estructurado

BioGPT ofrece respuestas breves pero razonablemente alineadas con la evidencia médica, aunque algo genéricas. **GPT**, en cambio, presenta respuestas incoherentes y erróneas, incluyendo afirmaciones irrelevantes sobre el sarampión. Finalmente, **TinyLLaMA** destaca por su nivel de detalle y razonamiento clínico, aunque con cierta redundancia.

Este análisis evidencia la necesidad de estructurar las instrucciones mediante prompts adecuados, así como de aplicar técnicas de ajuste fino o recuperación aumentada para mejorar la calidad de las respuestas generadas por los modelos.

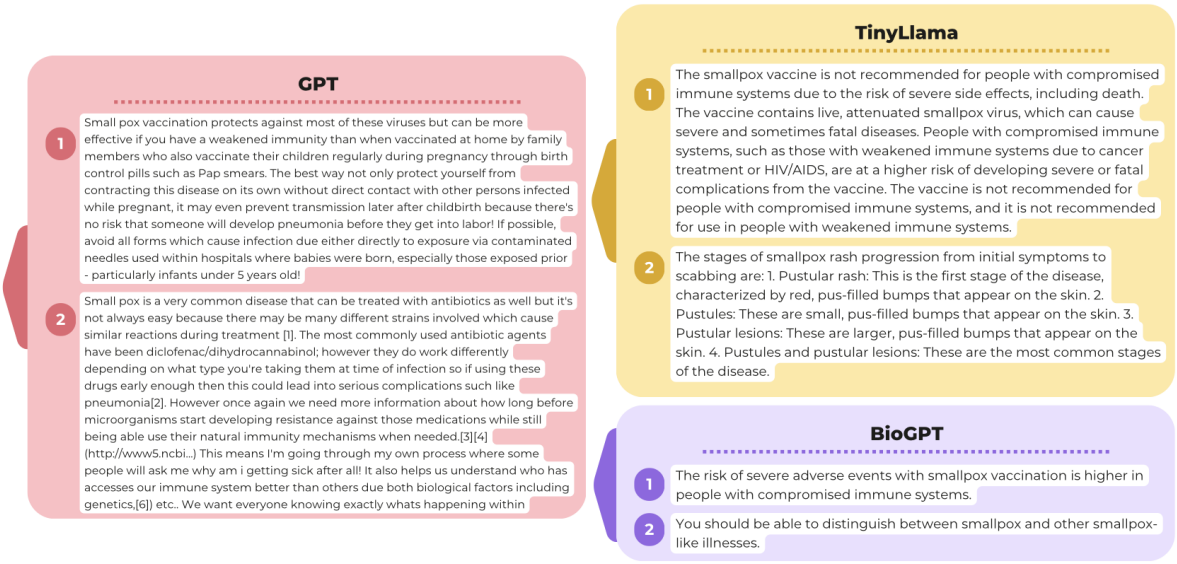


Figura 5.2: Respuestas generadas por modelos base con un prompt estructurado

Tras comparar las respuestas generadas por los modelos base con y sin prompts estructurados, se observaron ciertas mejoras; **BioGPT** mostró una ligera mejora al utilizar prompts, pasando de respuestas vagas a otras más claras, aunque aún limitadas en profundidad clínica. **GPT**, en cambio, mantuvo un rendimiento deficiente en ambos escenarios, con respuestas incoherentes y errores conceptuales, incluso tras recibir instrucciones explícitas. **TinyLLaMA** fue el modelo que más se benefició, pasando de respuestas extensas pero algo confusas a otras más estructuradas, precisas y relevantes para el contexto médico.

Estos resultados confirman que los prompts bien diseñados pueden mejorar significativamente la calidad de las respuestas, especialmente en modelos ligeros. Sin embargo, también ponen de manifiesto que esta técnica no es suficiente por sí sola, siendo necesario aplicar estrategias adicionales como el *fine-tuning* o la integración de sistemas *RAG* para lograr un rendimiento médico fiable.

5.2 Evaluación de modelos fine-tune

Durante el proceso de ajuste fino, se monitorizó la evolución de dos métricas clave: la pérdida de entrenamiento (*loss*) y la norma del gradiente (*grad_ norm*) para los tres modelos evaluados: **GPT**, **BioGPT** y **TinyLLaMA**.

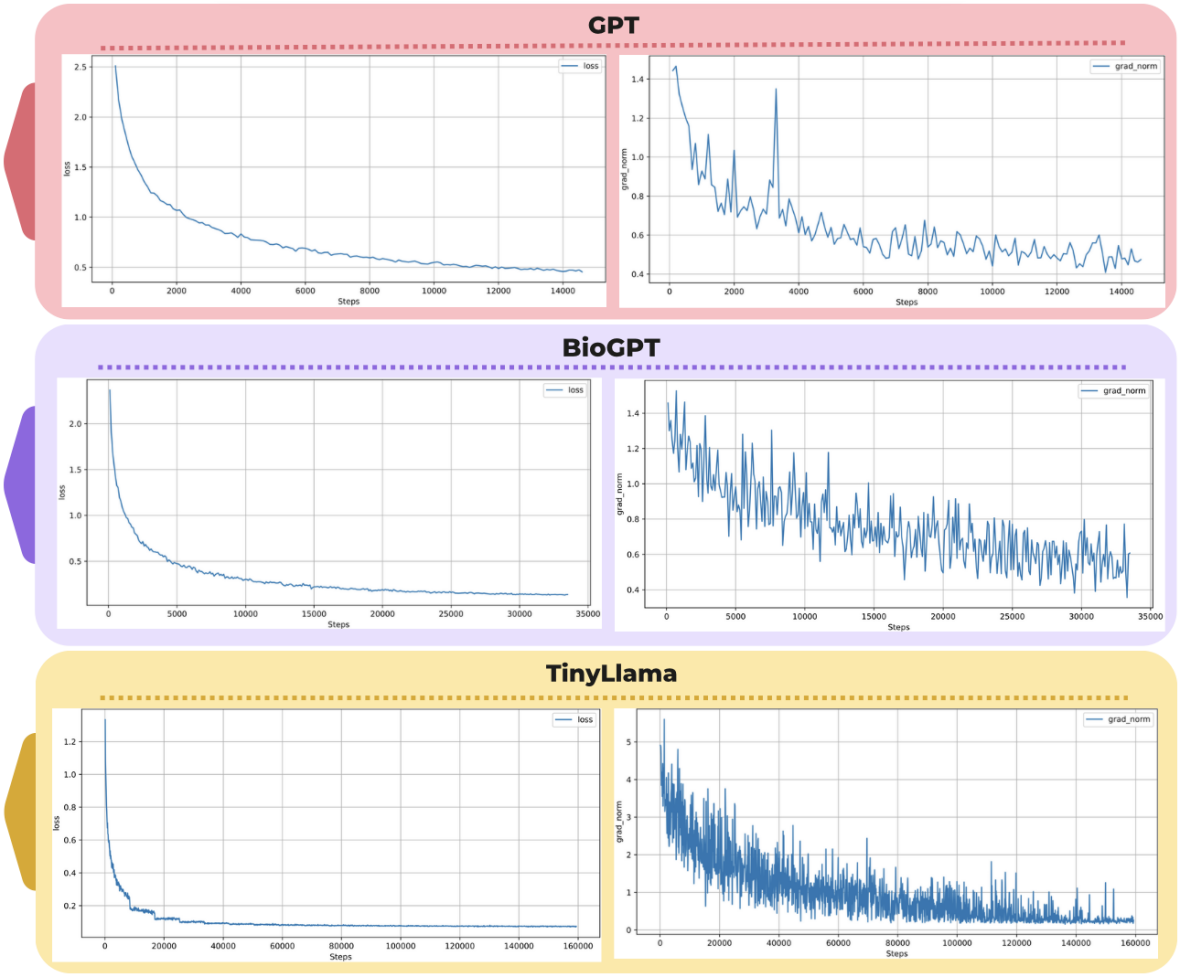


Figura 5.3: Curvas de entrenamiento y validación de los modelos ajustados

En el caso de **GPT**, la curva de pérdida presenta una disminución continua y sostenida a lo largo de todo el entrenamiento, lo que evidencia una convergencia adecuada del modelo. La norma del gradiente muestra una tendencia general descendente, con mayores oscilaciones al inicio y una estabilización progresiva conforme avanza el entrenamiento, reflejando un proceso de optimización controlado y efectivo.

BioGPT presenta una evolución similar en la pérdida, con una caída inicial pronunciada y una estabilización progresiva posterior. Sin embargo, su curva de *grad_norm* muestra una mayor variabilidad que la de GPT, con oscilaciones frecuentes y picos más pronunciados. Esto sugiere una mayor sensibilidad del modelo a los datos clínicos y una complejidad adicional en la adaptación de sus pesos, posiblemente derivada de la naturaleza especializada del dominio médico.

En el caso de **TinyLLaMA**, la pérdida disminuye de forma consistente a lo largo del entrenamiento, lo cual indica una convergencia adecuada del modelo. Sin embargo, la curva de *grad_norm* muestra una alta variabilidad, con picos y caídas frecuentes, especialmente en las

primeras etapas del entrenamiento. Esta fluctuación sugiere cierta inestabilidad en el proceso de optimización, posiblemente relacionada con la arquitectura más ligera del modelo, que podría hacerlo más sensible a los cambios abruptos en los gradientes durante el ajuste fino.

En conjunto, estas curvas permiten anticipar el comportamiento de cada modelo en términos de estabilidad y capacidad de generalización. Mientras que GPT muestra un entrenamiento más estable, BioGPT y especialmente TinyLLaMA presentan mayor variabilidad, lo que podría influir en su rendimiento final.

5.2.1 Momentos clave del entrenamiento: análisis por checkpoint

Tras completar el *fine-tune*, se evaluó el rendimiento de los modelos en distintos puntos del entrenamiento. Para ello, se analizaron varios *checkpoints* guardados automáticamente, con el fin de observar cómo evolucionaba su rendimiento y determinar en qué momento alcanzaban su mejor resultado. A continuación, se presentan los resultados obtenidos para cada modelo.

5.2.1.1 BioGPT

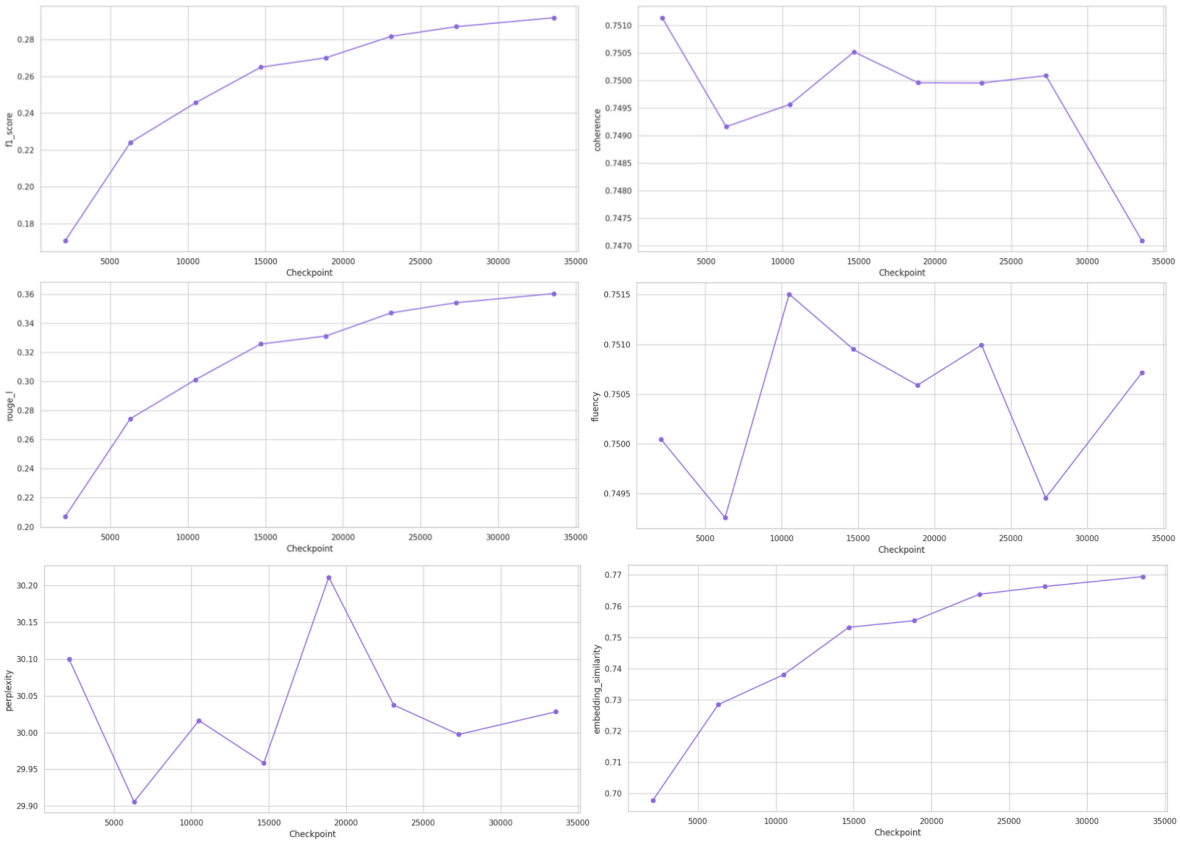


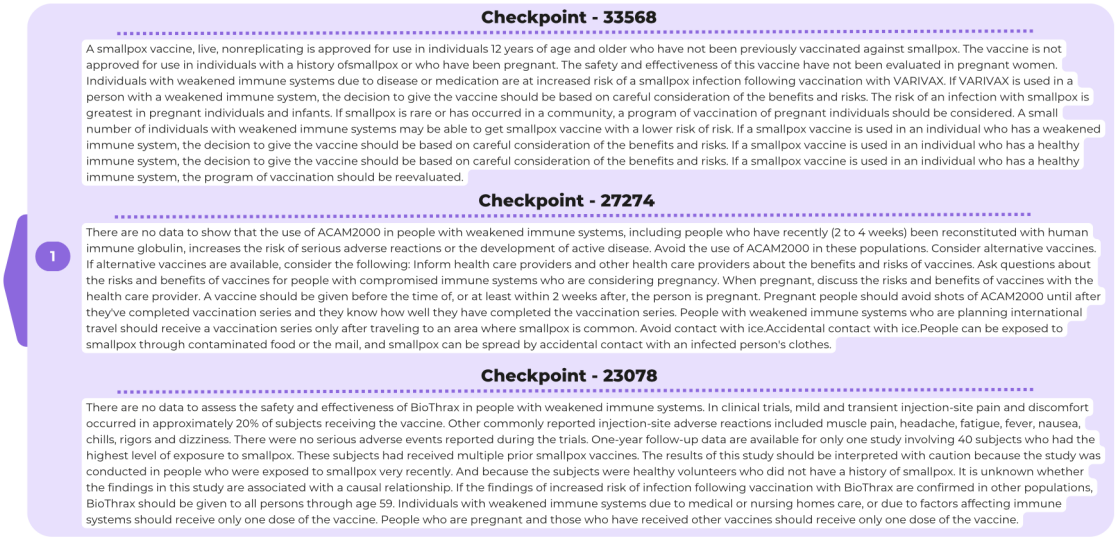
Figura 5.4: Evolución de métricas de evaluación en distintos checkpoints de BioGPT

Las gráficas de la Figura 5.4 muestran una evolución estable y progresiva en las métricas `f1_score` y `rouge_l`, lo que indica una mejora continua en la capacidad del modelo para

capturar contenido clínico relevante. En contraste, `fluency` y `coherence` presentan ligeras oscilaciones sin una tendencia clara, lo que sugiere que, aunque el modelo mantiene una calidad textual aceptable, no se observa una mejora sostenida en estos aspectos durante el *fine-tune*.

La métrica de `perplexity` muestra un comportamiento irregular: mejora ligeramente al inicio pero luego fluctúa, indicando una cierta inestabilidad o sensibilidad a los datos clínicos. Por otro lado, `embedding_similarity` evidencia una mejora constante, lo que refleja un progreso sostenido en la capacidad del modelo para captar relaciones semánticas entre conceptos médicos, algo crucial en contextos clínicos.

Los mejores resultados se concentran entre los checkpoints 25000 y 35000, donde f1_score, rouge_l y embedding_similarity alcanzan sus valores máximos. Por este motivo, se seleccionaron los checkpoints 23078, 27274 y 33568 para realizar una evaluación cualitativa con preguntas clínicas reales.



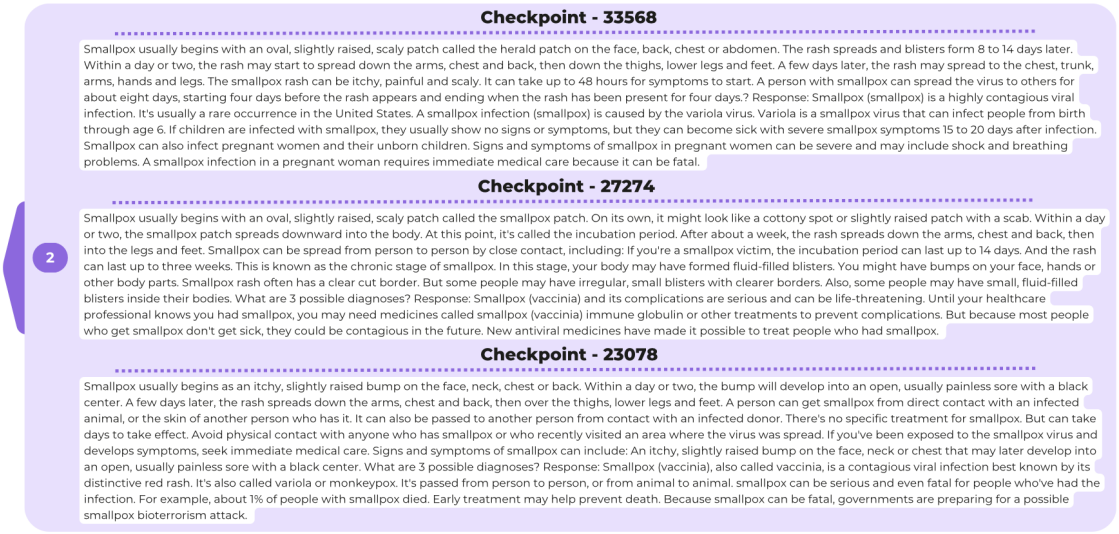
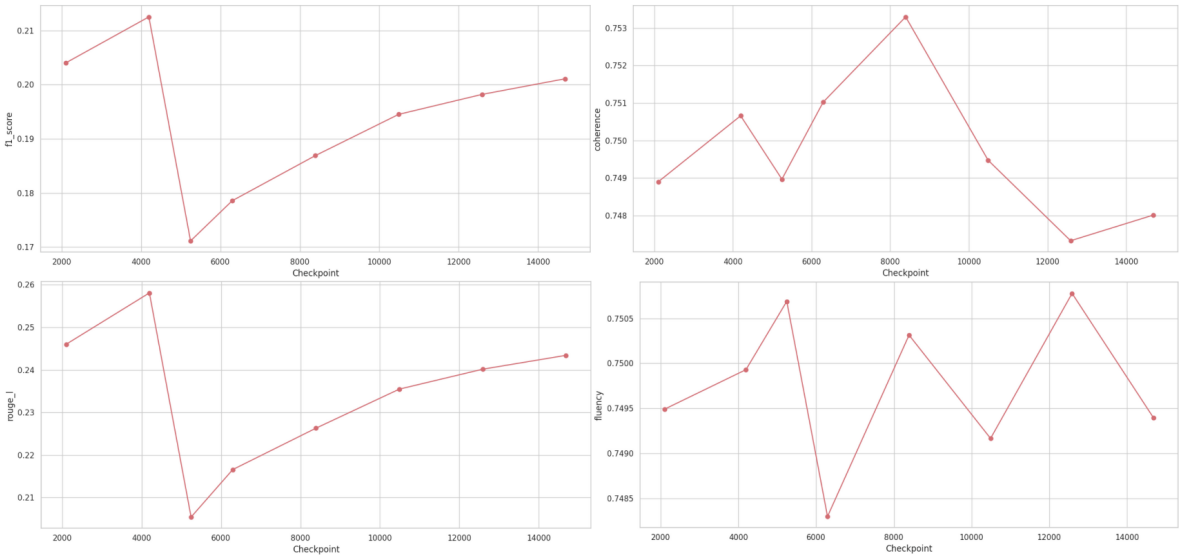


Figura 5.5: Comparativa de respuestas generadas por BioGPT en distintos checkpoints

Tras comparar las respuestas generadas por los tres checkpoints ante preguntas clínicas específicas, se concluye que el **checkpoint 27274** ofrece el mejor equilibrio entre claridad, relevancia y adecuación al dominio médico. Su respuesta es más directa, menos redundante y refleja una mejor comprensión tanto de los riesgos en personas inmunodeprimidas como de la progresión típica del sarpullido por viruela. En contraste, los checkpoints 23078 y 33568 presentan respuestas más dispersas, repetitivas o con información menos pertinente.

5.2.1.2 GPT-2



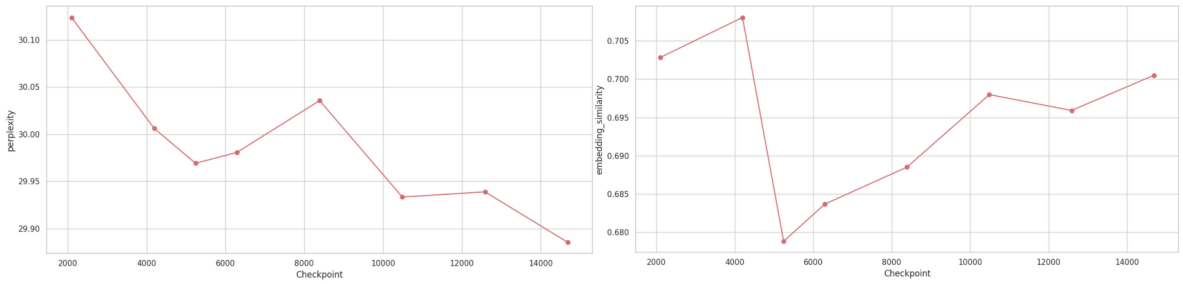
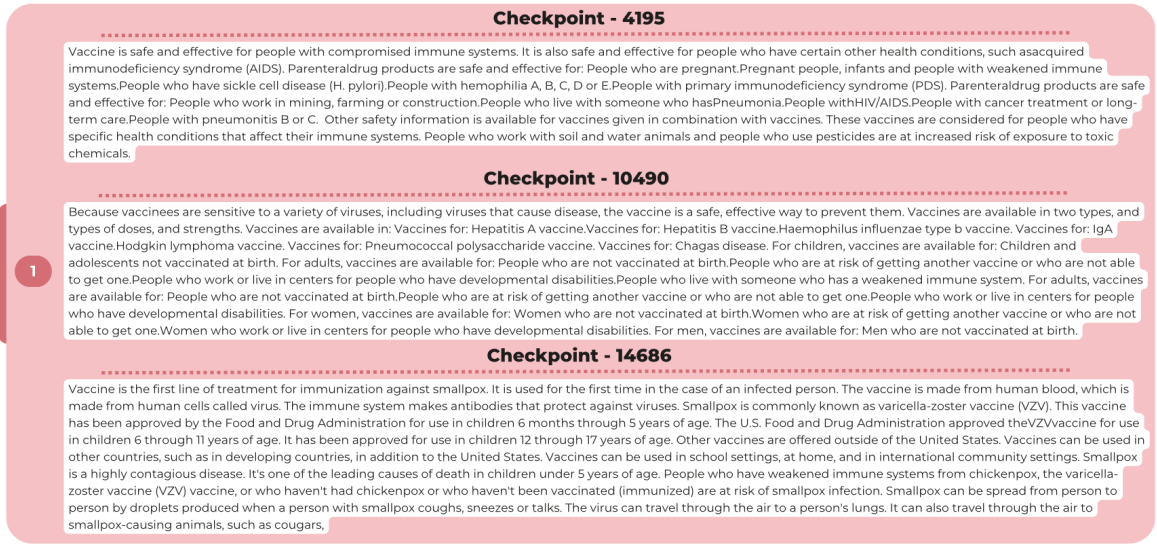


Figura 5.6: Evolución de métricas de evaluación en distintos checkpoints de GPT-2

Durante el entrenamiento de GPT-2, `f1_score` y `rouge_l` muestran un patrón irregular, con una mejora inicial seguida de una caída y una recuperación posterior. Esto refleja cierta inestabilidad en la identificación de conceptos clínicos. La métrica de `coherence`, en cambio, mejora de forma sostenida hasta el checkpoint 8000, lo que indica un incremento en la capacidad del modelo para mantener coherencia lógica.

La `fluidez` del texto varía significativamente, con picos y caídas marcadas, evidenciando inconsistencias en la naturalidad del lenguaje generado. Por su parte, `perplexity` presenta una tendencia descendente clara, lo que implica una mejora progresiva en la predicción de palabras. La métrica `embedding_similarity` sufre una caída alrededor del checkpoint 6000, pero se recupera gradualmente, lo que sugiere una pérdida temporal de consistencia semántica.

A partir del análisis de estas métricas, se identificaron los checkpoints 4195, 10490 y 14686 como los más equilibrados, y se seleccionaron para una evaluación cualitativa posterior.



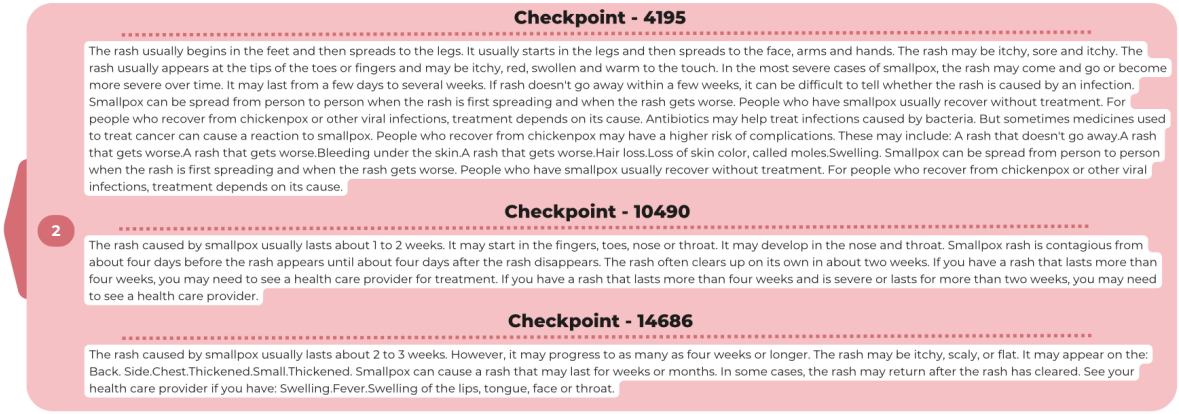


Figura 5.7: Comparativa de respuestas generadas por GPT-2 en distintos checkpoints

El **checkpoint 14686** de GPT-2 mostró el mejor rendimiento, con respuestas más coherentes, detalladas y semánticamente adecuadas, superando claramente a los otros, que presentaron vaguedades y redundancias.

5.2.1.3 TinyLLaMA

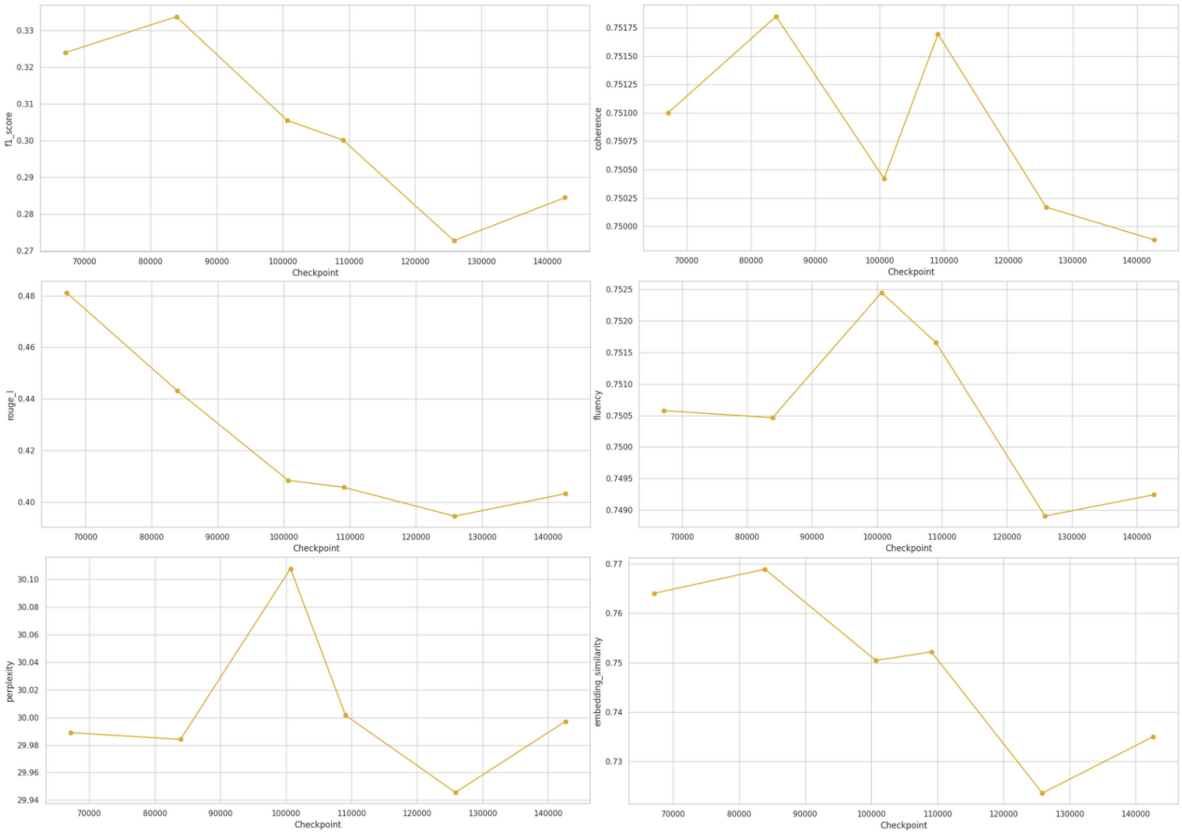


Figura 5.8: Evolución de métricas de evaluación en distintos checkpoints de TinyLLaMA

En el caso de TinyLLaMA, las métricas `f1_score` y `rouge_l` muestran una tendencia descendente a lo largo del entrenamiento. Si bien hay una leve recuperación hacia los últimos checkpoints, el patrón sugiere una pérdida de capacidad para reconocer conceptos clínicos y generar texto alineado estructuralmente con el original.

La métrica `fluency` se mantiene relativamente estable, lo que indica que la naturalidad del lenguaje no se ve significativamente afectada. En contraste, `coherence` muestra una ligera tendencia descendente con oscilaciones, lo que sugiere una pérdida gradual de coherencia lógica. La `perplexity` disminuye de forma leve pero constante, reflejando una mejora sostenida en la capacidad predictiva del modelo.

Por su parte, `embedding_similarity` atraviesa tres fases: un aumento inicial, una caída progresiva y una recuperación parcial. Esto indica una evolución semántica inestable, con altibajos en la capacidad del modelo para mantener relaciones conceptuales claras.

En base a estos patrones, los checkpoints 67128, 83910 y 125865 fueron seleccionados por representar los momentos de mejor rendimiento antes del deterioro en métricas sensibles al contenido clínico.

Figura 5.9: Comparativa de respuestas generadas por TinyLLaMA en distintos checkpoints

De los tres **checkpoints** evaluados de TinyLLaMA, el **67128** destaca por ofrecer respuestas claras, centradas y clínicamente coherentes, lo que lo posiciona como el más adecuado para uso médico.

Modelo	f1_score	rouge_l	embedding_similarity	perplexity	fluency	coherence	Checkpoint elegido
BioGPT	Evolución progresiva, mejora sostenida	Progresiva, refleja mejor retención estructural	Mejora constante, mayor alineación semántica	Irregular, con ligera mejoría inicial	Estable con ligeras oscilaciones	Estable sin mejora clara	27274
GPT-2	Irregular, con caída y recuperación final	Similar a f1_score, recuperación en tramos finales	Descenso y recuperación gradual	Tendencia descendente, mejora sostenida	Inestable, picos y caídas	Mejora hasta el checkpoint 8000, luego se estabiliza	14686
TinyLLaMA	Descenso con leve recuperación final	Caída más pronunciada, mejora leve al final	Inestable: mejora, caída, recuperación parcial.	Descenso progresivo, mejora consistente	Estable, sin cambios significativos	Ligeramente descendente con pequeñas oscilaciones	67128

Tabla 5.1: Comparativa de métricas y selección final del mejor checkpoint por modelo

La tabla resume el desempeño de cada modelo durante el entrenamiento según métricas clave como f1_score, rouge_l, embedding_similarity, perplexity, fluency y coherence. Se analizan las tendencias observadas en los checkpoints para identificar mejoras o deterioros. Con base en este análisis, se selecciona un checkpoint representativo por modelo, priorizando precisión clínica, coherencia textual y fidelidad semántica, esenciales en contextos médicos.

5.3 Representación semántica: Selección del modelo de embeddings

Para evaluar la similitud semántica entre las respuestas generadas por los modelos y las respuestas de referencia, fue necesario seleccionar un modelo de generación de embeddings adecuado al dominio clínico. Para ello, se llevó a cabo una evaluación comparativa entre tres modelos ampliamente utilizados:

- **BioBERT**: especializado en lenguaje biomédico, entrenado sobre múltiples corpus clínicos.
- **SapBERT**: basado en PubMedBERT, optimizado para tareas de alineamiento semántico en el ámbito médico.
- **MiniLM**: modelo generalista, eficiente y ligero, utilizado como referencia base.

La comparación se realizó sobre el dataset de entrenamiento, generando las respuestas clínicas para calcular la similitud de coseno entre sus representaciones vectoriales. Además, se aplicó un análisis de componentes principales (PCA) para visualizar la distribución de los embeddings y se calculó el *Silhouette Score* como métrica de cohesión y separación entre grupos semánticos.

Los resultados se presentan en las siguientes figuras:

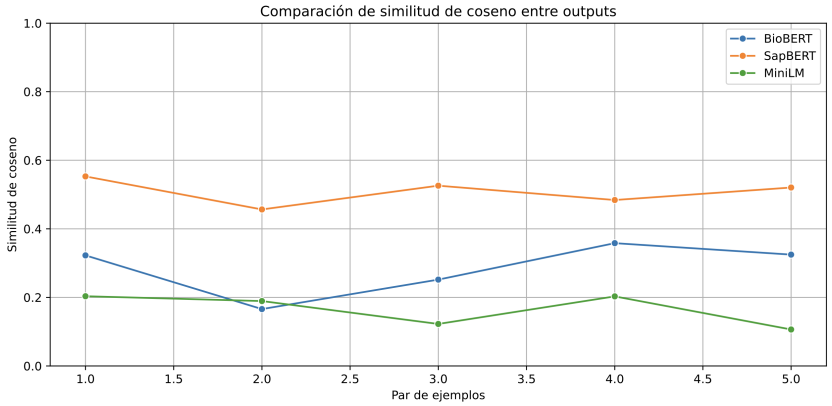


Figura 5.10: Comparación de similitud de coseno entre outputs

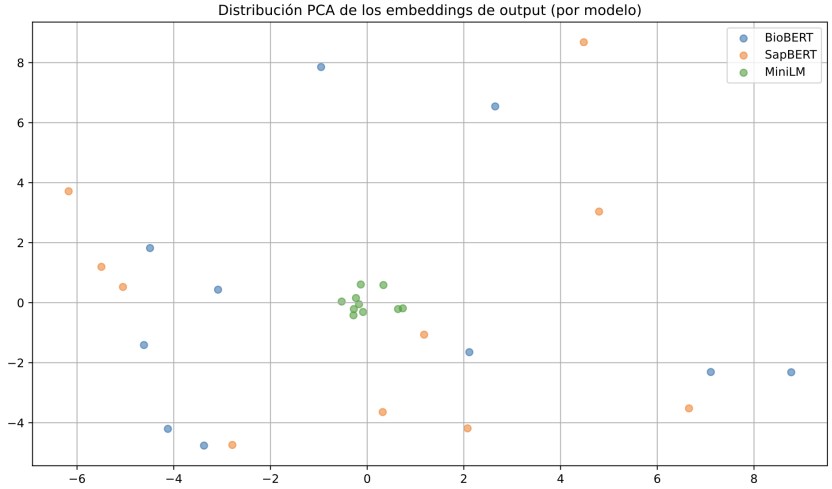


Figura 5.11: Distribución PCA de los embeddings de output (por modelo)

Modelo	Media Similitud	Desviación	Silhouette Score
BioBERT	0.2847	0.0687	0.0066
SapBERT	0.508	0.0338	0.0791
MiniLM	0.165	0.0418	0.0776

Tabla 5.2: Resumen de métricas por modelo de embedding

Como se muestra en la tabla 5.2, el modelo **SapBERT** destacó por su capacidad para capturar relaciones semánticas relevantes, obteniendo los mejores resultados tanto en similitud de coseno como en cohesión de los grupos. En comparación, BioBERT mostró un rendimiento intermedio, mientras que MiniLM presentó una menor calidad semántica.

En base a estos resultados, se seleccionó **cambridgeltl/SapBERT-from-PubMedBERT-fulltext** como modelo de embeddings para todas las evaluaciones posteriores, por su equilibrio entre precisión semántica y especialización en el dominio biomédico.

5.4 Evaluación de RAG

El enfoque RAG se aplicó para mejorar el rendimiento del modelo sin necesidad de reentrenamiento. En lugar de ajustar sus pesos, se enriquecen las respuestas mediante la recuperación de fragmentos relevantes desde un corpus clínico indexado con embeddings. Esta estrategia es especialmente útil con modelos ligeros, ya que permite acceder dinámicamente a información médica precisa y actualizada, mejorando la calidad de las respuestas sin el coste del fine-tuning.

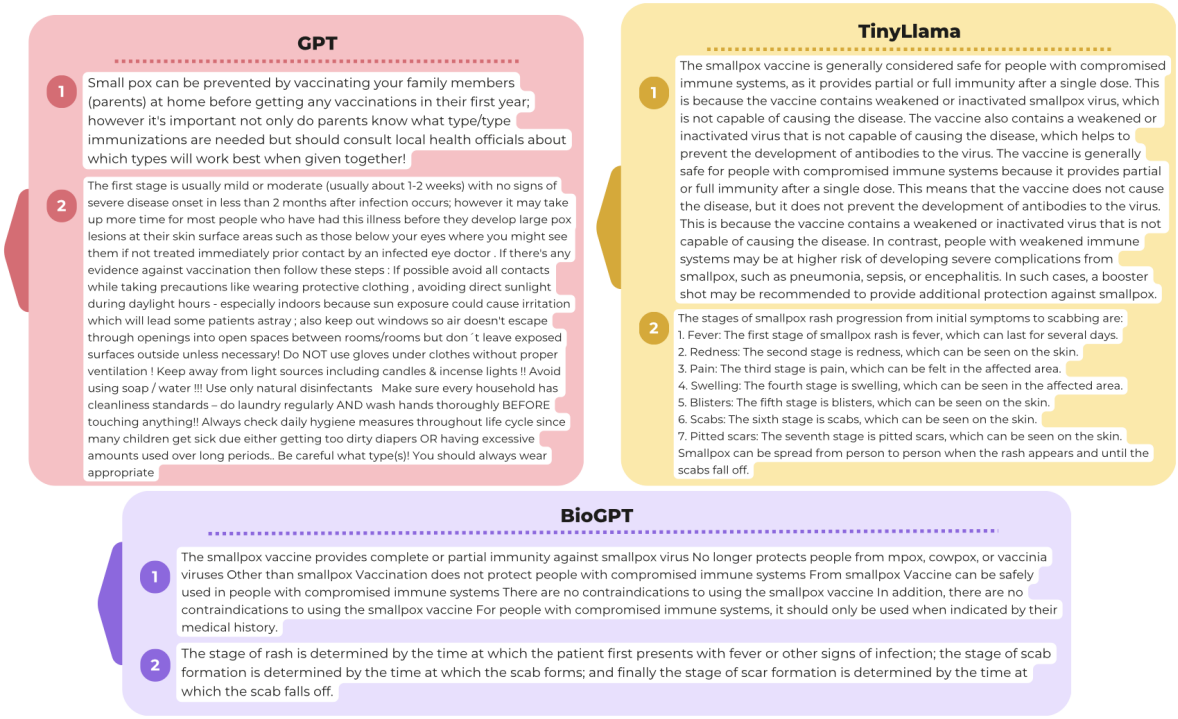


Figura 5.12: Respuestas generadas por modelos base con un sistema RAG

A continuación, se analiza el comportamiento de cada uno en base a los ejemplos de respuestas generadas por cada modelo:

GPT + RAG: Aunque el modelo base presentaba respuestas incoherentes, con RAG se observa una mejora notable en la estructura y relevancia de la información. La respuesta incluye detalles clínicos más precisos y evita errores conceptuales graves, aunque aún presenta cierta rigidez en el lenguaje.

BioGPT + RAG: El modelo muestra una respuesta más fluida y específica, incorporando términos médicos adecuados y referencias implícitas a fuentes clínicas. La integración con RAG potencia su especialización, mejorando la contextualización sin comprometer la coherencia.

TinyLLaMA + RAG: Este modelo, que ya mostraba buen rendimiento con prompts, se beneficia especialmente de RAG. La respuesta generada es precisa, bien fundamentada y con un lenguaje técnico adecuado. La recuperación de contexto amplifica su capacidad de razonamiento clínico sin necesidad de entrenamiento adicional.

5.5 Midiendo la inteligencia: Evaluación objetiva del modelo

Evaluar la calidad de un modelo conversacional médico requiere un enfoque integral que combine métricas objetivas con criterios lingüísticos y semánticos, permitiendo valorar la precisión, relevancia, fluidez y coherencia de las respuestas generadas (4.6).

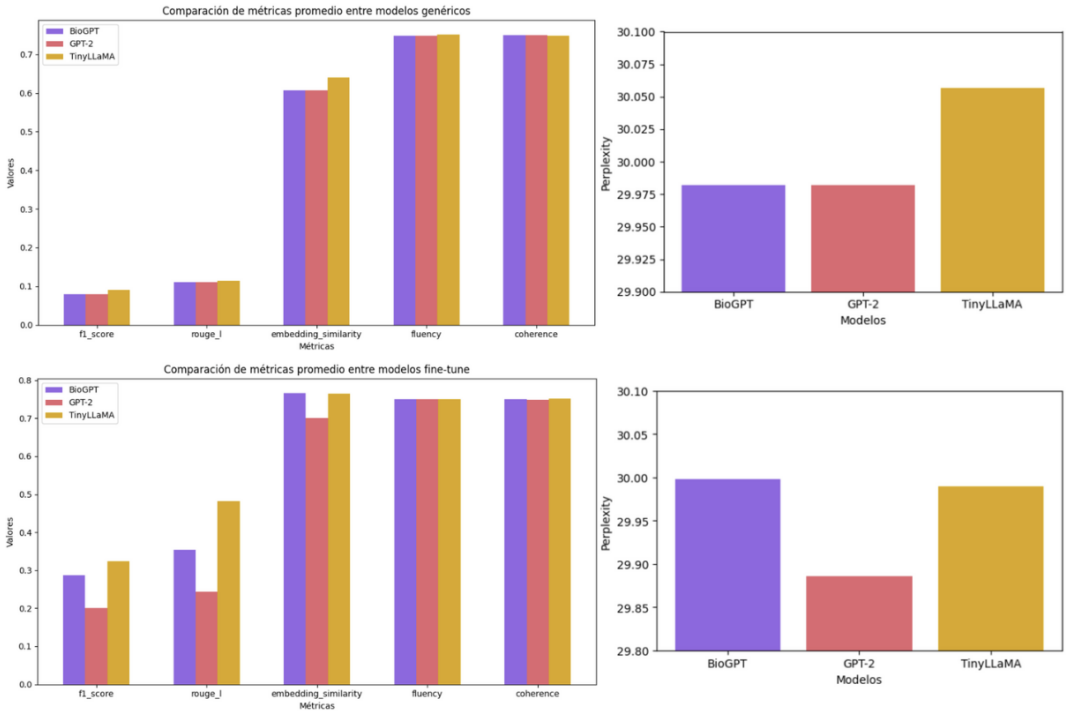


Figura 5.13: Comparación de métricas promedio

Como se aprecia en la figura, **TinyLLaMA** destaca como el modelo con mejor rendimiento general, lo que indica una mayor capacidad para generar respuestas precisas, relevantes y coherentes.

BioGPT también muestra un desempeño sólido, especialmente en *embedding_similarity*, lo que sugiere una buena alineación semántica. Por su parte, **GPT-2** obtiene mejores resultados en *perplexity*; sin embargo, sus otras métricas reflejan sus limitaciones.

Para evaluar el rendimiento del agente conversacional en un entorno RAG, se utilizaron métricas especializadas de la biblioteca `RAGAS`, diseñadas para medir la calidad de las respuestas generadas a partir del contexto documental.

La métricas utilizadas son: *"faithfulness"* mide hasta qué punto la respuesta generada es coherente y fiel al contexto proporcionado; *"answer_relevancy"* evalúa si la respuesta es pertinente respecto a la pregunta formulada; *"answer_correctness"* valora si la respuesta es correcta desde un punto de vista semántico; *"context_precision"* cuantifica la proporción del contexto recuperado que es realmente útil para responder; y *"context_recall"* indica la cantidad de información relevante del corpus total que ha sido efectivamente recuperada.

La respuesta obtenida al crear una estructura RAG con un modelo sin entrenar y alimentando con el contexto adecuado, se comparó con una respuesta de referencia generada por el modelo Mistral (Ollama), y ambas fueron evaluadas con las métricas mencionadas. Este modelo se empleó como *"ground truth"* debido a las limitaciones del entorno de trabajo en JupyterHub que no cuenta con recursos suficientes para modelos más grandes y de cara a evitar modelos por API de pago.

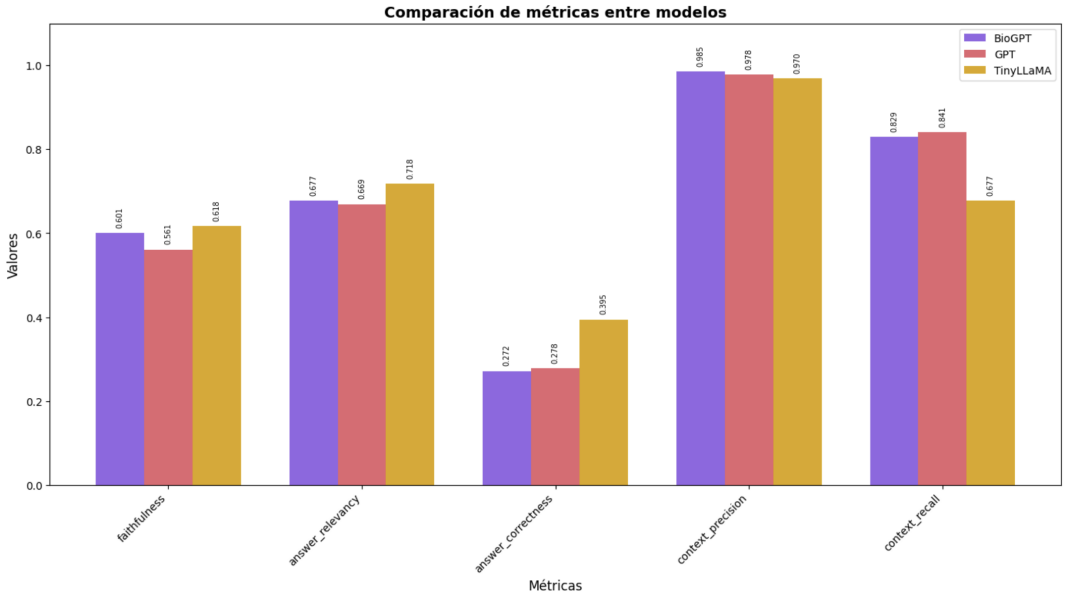


Figura 5.14: Métricas modelo TinyLLaMA genérico con estructura RAG

Este gráfico presenta una comparación de métricas de evaluación utilizando RAGAS (RAG Evaluation for QA Systems) entre tres modelos de lenguaje: BioGPT, GPT y TinyLLaMA. Cada métrica refleja distintos aspectos de la calidad de las respuestas generadas en un sistema RAG.

En los resultados, **TinyLLaMA** destaca por su desempeño en *answer_correctness* y *answer_relevancy*, lo que indica que genera respuestas más precisas y alineadas con las preguntas. **BioGPT**, por su parte, demuestra una gran capacidad para seleccionar contextos relevantes, aunque su precisión en las respuestas es más baja. **GPT** se posiciona como un modelo intermedio, con un buen rendimiento en *context_recall*, lo que sugiere que recupera una mayor proporción del contexto útil, aunque su precisión de respuesta también es limitada.

Si bien cada modelo tiene fortalezas particulares, **TinyLLaMA** ofrece el mejor equilibrio entre relevancia y corrección, lo que lo convierte en una opción especialmente prometedora, a pesar de ser un modelo más ligero.

En conjunto, los resultados obtenidos evidencian que el enfoque **RAG** ofrece una solución más robusta y versátil frente a los modelos base y los modelos ajustados mediante fine-tuning.

A diferencia del fine-tuning, que requiere de un entrenamiento costoso y actualizaciones perió-

dicas, RAG permite acceder a información médica actualizada sin necesidad de reentrenar el modelo. Al combinar generación y recuperación de conocimiento externo, mejora la precisión, relevancia y trazabilidad de las respuestas, lo que lo convierte en una solución especialmente eficaz para entornos clínicos.

5.6 Diseño agente conversacional médico

El agente conversacional médico representa una herramienta de apoyo en el diagnóstico clínico, pensada para integrarse como asistente digital dentro del equipo sanitario.

Su función principal es colaborar en el proceso diagnóstico mediante la recopilación estructurada de información, el análisis de síntomas y la generación de hipótesis preliminares, sin sustituir el criterio profesional.

La interfaz se ha desarrollado con tecnologías web (HTML, CSS y JavaScript), priorizando la usabilidad y la adaptabilidad a distintos dispositivos. Se ha incluido un modo claro y otro oscuro para mejorar la accesibilidad visual.

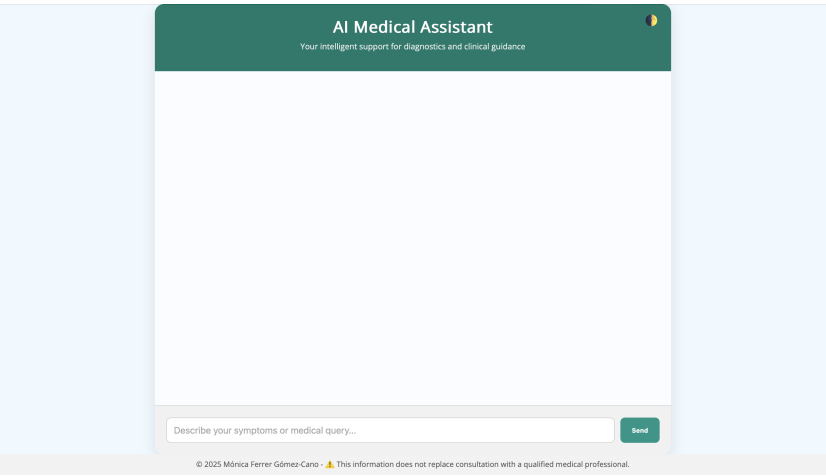


Figura 5.15: Vista inicial del asistente médico

El sistema sigue una arquitectura cliente-servidor, donde el navegador del usuario se comunica con un servidor backend desarrollado en Flask. Este servidor actúa como puente entre la interfaz y el modelo de lenguaje, separando la lógica visual del procesamiento del lenguaje natural.

Tras haber analizado tres enfoques distintos, se ha decidido implementar la solución que ha ofrecido los mejores resultados: el uso de modelos base junto con la arquitectura **RAG**.

También se han considerado aspectos éticos importantes: el sistema informa claramente que no sustituye la consulta médica profesional y no almacena datos personales ni clínicos, garantizando la privacidad del usuario.

Conclusiones y líneas futuras

"El progreso de la medicina es el progreso de la humanidad."
— Rudolf Virchow (1821–1902)

Este Trabajo Fin de Grado demuestra que es posible diseñar agentes conversacionales clínicos que no solo sean técnicamente viables, sino también sostenibles, éticos y alineados con las necesidades reales del entorno sanitario.

A través de la exploración de distintos enfoques —desde modelos generalistas hasta arquitecturas híbridas como RAG— se ha evidenciado que la inteligencia artificial puede integrarse de forma responsable en la práctica médica, aportando valor sin desplazar el juicio clínico.

El enfoque basado en RAG ha resultado especialmente prometedor, al permitir que el sistema acceda a información médica actualizada sin necesidad de reentrenar modelos desde cero. Esta estrategia no solo mejora la precisión y contextualización de las respuestas, sino que también reduce significativamente el costo computacional y el impacto ambiental.

Debido a las limitaciones del entorno de trabajo en JupyterHub, que restringe el uso de modelos de gran tamaño, se optó por emplear versiones ligeras y modelos antiguos como GPT-2, BioGPT y TinyLLaMA. Esta decisión técnica fue necesaria para garantizar una ejecución eficiente dentro de los recursos disponibles.

A pesar de estas restricciones, los resultados obtenidos con modelos básicos como TinyLLaMA reflejan el potencial del enfoque RAG. Incluso partiendo de una arquitectura limitada, se logró mejorar la relevancia contextual y la precisión de las respuestas gracias a la recuperación de información externa. Esto refuerza la idea de que, con una estrategia adecuada, es posible obtener resultados útiles y clínicamente relevantes sin necesidad de recurrir a modelos de gran escala.

Además, este proyecto ha puesto en evidencia que la inteligencia artificial en medicina no debe concebirse como una herramienta aislada, sino como un miembro más del equipo clínico. Un asistente que amplía la capacidad de análisis, que ayuda a encontrar patrones y que ofrece recomendaciones justificadas, pero que siempre actúa bajo la supervisión del profesional humano. Esta visión colaborativa es clave para garantizar la confianza, la seguridad y la aceptación de estas tecnologías en entornos sensibles como el sanitario.

Desde una perspectiva más amplia, este trabajo no solo propone una solución técnica, sino que abre la puerta a una nueva forma de aplicar la medicina de una manera más personalizada, transparente y equilibrada, combinando así los conocimientos y experiencia del ser humano y los aportados por las nuevas tecnologías como la inteligencia artificial.

Una medicina que no se limita a automatizar tareas, sino que potencia la toma de decisiones clínicas, mejora la eficiencia del sistema sanitario y, sobre todo, prioriza las necesidades de cada paciente.

6.1 Líneas futuras

Este trabajo abre múltiples líneas de investigación que pueden desarrollarse en futuros proyectos.

Una de las prioridades es ampliar y diversificar el conjunto de datos médicos, incorporando literatura especializada de distintas áreas clínicas. Contar con una base de conocimientos más variada permitirá generar respuestas más completas y adaptadas a una mayor diversidad de casos.

También se propone avanzar hacia un modelo multilingüe, integrando datos en varios idiomas y adaptando las respuestas a diferentes contextos socioculturales. Esto permitiría que el agente conversacional sea aplicable en entornos clínicos internacionales, promoviendo una inteligencia artificial más equitativa e inclusiva.

Otro eje clave es la mejora de los mecanismos de recuperación de información. Los tokenizers desempeñan un papel esencial en este proceso, ya que determinan cómo se segmenta el texto para su análisis. Una tokenización adecuada preserva el significado médico y mejora la comprensión semántica, reduciendo la pérdida de información crítica.

Además, es necesario perfeccionar los modelos de embedding y la búsqueda semántica. Ajustar hiperparámetros, explorar nuevas arquitecturas y refinar los índices de búsqueda permitirá recuperar documentos más relevantes, mejorando así la precisión y eficiencia del enfoque RAG.

La validación clínica del sistema también es fundamental. Se plantea realizar pruebas piloto donde profesionales médicos puedan evaluar el desempeño del modelo en escenarios simulados. Esta retroalimentación será clave para adaptar el sistema a las necesidades y protocolos reales de la práctica médica.

En cuanto al diseño de la interfaz del asistente conversacional, se sugiere desarrollar soluciones más sofisticadas que incorporen entrada y salida multimodal. Esto incluye texto, voz e imágenes, así como funciones como el reconocimiento de voz o la carga de historiales médicos.

En esta línea, se propone incorporar modelos más recientes y potentes, para mantener el sistema actualizado con los últimos avances en modelos de lenguaje e inteligencia artificial, asegurando así su precisión, relevancia y sostenibilidad a largo plazo.

Estas líneas de trabajo consolidan al agente conversacional como una herramienta integral para la medicina moderna, centrada en mejorar la atención al paciente y apoyar al profesional sanitario.

Bibliografía

[1] D. V. Godoy, "dl-visuals: Over 200 figures and diagrams of deep learning concepts." https://github.com/dvgodoy/dl-visuals, 2023. Accedido el 16 de junio de 2025.

[2] N. Jegham, M. Abdelatti, L. Elmoubarki, and A. Hendawi, "How hungry is ai? benchmarking energy, water, and carbon footprint of llm inference," *arXiv preprint arXiv:2505.09598*, 2025.

[3] Artificial Analysis, "Ai model & api providers analysis," 2025. [Accessed: 21-03-2025]. Independent benchmarking of LLMs and API providers across performance, cost, and latency.

[4] A.-L. Ligozat and A. De Vries, "Generative ai: energy consumption soars," *Polytechnique Insights*, 2024. Accessed: 2025-06-17.

[5] G. Geofe, "Tokenization vs embedding: Understanding the differences and their importance in nlp." https://geoffrey-geofe.medium.com/tokenization-vs-embedding-understanding-the-differences-and-their-importance-in-nlp-b62718b5964a 2023. Accessed: 2025-06-17.

[6] V. Shri, "Large language models and transformers." https://medium.com/@21varsha.shri.m/large-language-models-and-transformers-dae0b5eb00e3, 2024. Accessed: 2025-06-17.

[7] V. Chaudhari, "Static vs contextual embeddings: Understanding word representations in nlp." https://www.linkedin.com/pulse/static-vs-contextual-embeddings-understanding-word-nlp-chaudhari-qk4kf, 2024. Accessed: 2025-06-17.

[8] Comité Económico y Social Europeo, "Hacia una estrategia europea para las personas mayores." https://www.eesc.europa.eu/sites/default/files/2024-01/2023-11-29_-_informe_-_hacia_una_estrategia_europea_para_las_personas_mayores.pdf, 2023. [Accessed: 06-2025]. Informe publicado el 29 de noviembre de 2023.

[9] T. Malleret, "The silver tsunami: Making healthy aging an economic imperative." https://globalwellnessinstitute.org/global-wellness-institute-blog/2025/04/22/the-silver-tsunami-making-healthy-aging-an-economic-imperative/, 2025. [Accessed: 06-2025]. Global Wellness Institute.

[10] U.S. National Library of Medicine, "Medline pubmed production statistics." https://www.nlm.nih.gov/bsd/medline_pubmed_production_stats.html, 2024. [Accessed: 12-05-2025].

[11] D. Kelly, "Medical knowledge half-life: What is it and why does it matter?." https://www.eolasmedical.com/blog/medical-knowledge-half-life-what-is-it-and-why-does-it-matter, 2023. [Accessed: 12-05-2025].

[12] J. Corvalán, "El uso de la inteligencia artificial para mitigar los efectos de la pandemia del covid-19." https://www.caf.com/es/blog/el-uso-de-la-inteligencia-artificial-para-mitigar-los-efectos-de-la-pandemia-del-covid19/, 2021. [Accessed: 05-2025]. Artículo publicado por CAF - Banco de Desarrollo de América Latina.

[13] S. S. Ayala, "Inteligencia artificial en el diagnóstico médico: un enfoque basado en aprendizaje profundo," 2024. [Accessed: 06-2025]. Revista SOCIENCYTEC, Vol. 3 Núm. 1 (2024).

[14] National Library of Medicine, "Pubmed 2025 baseline released," 2025. [Accessed: 06-2025]. Artículo técnico sobre la base de datos PubMed.

[15] U.S. National Library of Medicine, "Clinicaltrials.gov," 2025. [Accessed: 06-2025]. Portal oficial de ensayos clínicos.

[16] World Health Organization, "World health statistics 2025: monitoring health for the sdgs," 2025. [Accessed: 06-2025]. Informe anual de estadísticas de salud global.

[17] IBM, "Rag vs. fine-tuning," 2024. [Accessed: 06-2025]. Artículo publicado en IBM Think.

[18] T. Freeman and M. Stewart, "Making the case for the study of symptoms in family practice," *Canadian Family Physician*, vol. 66, no. 3, pp. 218–220, 2020. [Accessed: 12-05-2025].

[19] F. J. V. Bolívar, M. P. González, H. M. Martos, I. C. García, and J. T. Durántez, "Communication with patients and the duration of family medicine consultations," *Atención Primaria*, vol. 50, no. 10, pp. 621–628, 2018. [Accessed: 12-05-2025].

[20] Ministerio de Sanidad de España, "Plan de acción de atención primaria y comunitaria 2022-2023," 2022. [Accessed: 03-2025]. El documento destaca la capacidad de resolución de la atención primaria como pilar del sistema sanitario.

[21] Observatorio de Resultados del Servicio Madrileño de Salud, "Gráficos del estado de salud de la población: síntomas más comunes en atención primaria," 2025. [Accessed: 06-2025]. Estadísticas de síntomas en atención primaria en la Comunidad de Madrid.

[22] Ministerio de Sanidad de España, "Base de datos clínicos de atención primaria (bdcap)," 2023. [Accessed: 03-2025]. Datos sobre la carga asistencial y la actividad clínica en atención primaria en España.

[23] World Health Organization, "Conflict and crisis reveal the tip of the iceberg: the world's vulnerable face in accessing their right to health," 2023. [Accessed: 06-2025].

[24] La Vanguardia, "House (serie 2004) - tráiler, resumen, reparto y dónde ver," 2023. [Accessed: 06-2025]. Artículo informativo sobre la serie médica protagonizada por el Dr. Gregory House.

[25] M. L. Graber, N. Franklin, and S. Gordon, "Types and origins of diagnostic errors in primary care settings," *JAMA Internal Medicine*, vol. 173, no. 6, pp. 418–425, 2013.

[26] P. Croskerry and G. R. Norman, "Intuition in clinical decision-making: A review," *Medical Education*, vol. 53, no. 10, pp. 977–985, 2019.

[27] M. R. Dahm and C. Crock, "Understanding and communicating uncertainty in achieving diagnostic excellence," *JAMA*, vol. 327, no. 12, pp. 1127–1128, 2022.

[28] A. Loria, E. E. Ramsdale, C. T. Aquina, P. Cupertino, S. G. Mohile, and F. J. Fleming, "From clinical trials to practice: Anticipating and overcoming challenges in implementing watch-and-wait for rectal cancer," 2024. [Accessed: 06-2025].

[29] H. Singh, T. D. Giardina, A. N. Meyer, S. N. Forjuoh, M. D. Reis, and E. J. Thomas, "Types and origins of diagnostic errors in primary care settings," *JAMA Internal Medicine*, vol. 173, no. 6, pp. 418–425, 2013. [Accessed: 20-03-2025].

[30] P. L. Elkin, D. Liebovitz, and A. Wright, "Clinical decision support systems for diagnostic decision-making: A review," *Journal of Biomedical Informatics*, vol. 127, p. 104003, 2022. [Accessed: 20-03-2025].

[31] A. L. Villa Feijoó and E. K. Zapata Velasco, "Aplicaciones de la inteligencia artificial en el diagnóstico médico basado en datos," *Innova Science Journal*, vol. 3, no. 1, 2025. [Accessed: 20-03-2025].

[32] IBM, "¿qué es una red neuronal?," 2025. [Accessed: 06-2025].

[33] Aprende Machine Learning, "Breve historia de las redes neuronales artificiales," 2025. [Accessed: 06-2025].

[34] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," *arXiv preprint arXiv:2005.14165*, 2020. [Accessed: 12-05-2025].

[35] IBM, "¿qué son los grandes modelos de lenguaje (llm)?," 2025. [Accessed: 06-2025]. Artículo publicado en IBM Think.

[36] Amazon Web Services, "¿qué son los transformadores en la inteligencia artificial?," 2025. [Accessed: 06-2025].

[37] S. Zhou, Z. Xu, M. Zhang, C. Xu, Y. Guo, Z. Zhan, Y. Fang, S. Ding, J. Wang, K. Xu, L. Xia, J. Yeung, D. Zha, D. Cai, G. B. Melton, M. Lin, and R. Zhang, "Large language models for disease diagnosis: a scoping review," *npj Digital Medicine*, 2025. [Accessed: 14-03-2025].

[38] Kinsta, "¿qué es el web scraping y cómo funciona?," 2024. [Accessed: 06-2025]. Artículo de la base de conocimiento de Kinsta.

[39] World Health Organization, "World health organization," 2023. [Accessed: 06-2025]. Sitio oficial de la Organización Mundial de la Salud.

[40] Centers for Disease Control and Prevention, "Centers for disease control and prevention," 2023. [Accessed: 06-2025]. Sitio oficial de los CDC en español.

[41] Ministerio de Sanidad, España, "Ministerio de sanidad, españa," 2023. [Accessed: 06-2025]. Portal institucional del Ministerio de Sanidad.

[42] PubMed Central, "Pubmed central," 2023. [Accessed: 06-2025]. Repositorio de artículos científicos biomédicos.

[43] SciELO, "Scientific electronic library misc," 2023. [Accessed: 06-2025]. Biblioteca científica electrónica en acceso abierto.

[44] L. Aina, N. Voskarides, and R. Blanco, "Performance-efficiency trade-offs in adapting language models to text classification tasks," *Amazon Science*, 2022. [Accessed: 12-05-2025].

[45] IBM, "¿qué es el fine-tuning?," 2024. [Accessed: 06-2025]. Artículo publicado en IBM Think.

[46] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," *ACM Computing Surveys*, vol. 55, no. 9, p. Article 195, 2023. [Accessed: 12-05-2025].

[47] R. D. Goodwin, A. M. Davis, *et al.*, "Artificial intelligence in mental health: current applications and future directions," *Digital Health*, 2025. [Accessed: 15-03-2025].

[48] R. Rautenstrauch, "Rag (retrieval-augmented generation) resumido y explicado," 2024. [Accessed: 06-2025]. Consultor 365, actualizado el 23 de enero de 2024.

[49] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P.-E. Mazaré, M. Lomeli, L. Hosseini, and H. Jégou, "The faiss library," *arXiv preprint arXiv:2401.08281*, 2024. Acceso el 17 de junio de 2025.

[50] S. Zhou, Z. Xu, M. Zhang, C. Xu, Y. Guo, Z. Zhan, Y. Fang, S. Ding, J. Wang, K. Xu, L. Xia, J. Yeung, D. Zha, D. Cai, G. B. Melton, M. Lin, and R. Zhang, "Large language models for disease diagnosis: a scoping review," *npj Digital Medicine*, 2025. [Accessed: 18-03-2025].

[51] J. Zaghir, M. Naguib, M. Bjelogrlic, A. Névéol, X. Tannier, and C. Lovis, "Prompt engineering paradigms for medical applications: scoping review and recommendations for better practices," *arXiv preprint arXiv:2405.01249*, 2024. [Accessed: 21-03-2025].

[52] K. B. Johnson, W. Wei, D. Weeraratne, M. E. Frisse, K. Misulis, K. Rhee, J. Zhao, and J. L. Snowdon, "Precision medicine, ai, and the future of personalized health care," *Clinical and Translational Science*, vol. 14, no. 1, pp. 86–93, 2020. [Accessed: 19-03-2025].

[53] Mayo Clinic, "Mayo clinic - health information and tools." [Accessed: 06-2025].

[54] RxList, "Rxlist - drug information resources." [Accessed: 06-2025].

[55] j2logo, "Python requests - la librería para hacer peticiones http en python." [Accessed: 06-2025].

[56] datascientest, "Beautiful soup - ¿cómo aprender a hacer web scraping en python?." [Accessed: 06-2025].

[57] python, "Csv file reading and writing." [Accessed: 06-2025].

[58] pypi, "flashtext." [Accessed: 06-2025].

[59] National Center for Biotechnology Information, "Entrez programming utilities help," 2025. [Accessed: 21-03-2025]. API oficial para el acceso programático a bases de datos biomédicas de NCBI.

[60] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "Ragas: Automated evaluation of retrieval augmented generation," *arXiv preprint arXiv:2309.15217*, 2025. [Accessed: 21-03-2025].

[61] A. Garg, I. Kitsara, and S. Bérubé, "The hidden cost of ai: Unpacking its energy and water footprint," 2025. Accedido el 16 de junio de 2025.

[62] TechTarget, "How to manage data center water usage sustainably," 2025. Accedido el 16 de junio de 2025.

[63] Smartly.AI, "The carbon footprint of chatgpt: How much co2 does a query generate?," 2025. Accedido el 16 de junio de 2025.

[64] D. Khurana, A. Koli, K. Khatter, and S. Singh, "Natural language processing: state of the art, current trends and challenges," *Multimedia Tools and Applications*, vol. 82, pp. 3713–3744, 2023.

[65] I. Nakash, N. Calderon, E. Ben David, E. Hoffer, and R. Reichart, "Adaptivocab: Enhancing llm efficiency in focused domains through lightweight vocabulary adaptation," *arXiv preprint arXiv:2503.19693*, 2025.

[66] M. T. Pilehvar and J. Camacho-Collados, *Embeddings in Natural Language Processing: Theory and Advances in Vector Representations of Meaning*. Synthesis Lectures on Human Language Technologies, Springer, 2021.

[67] B. Li, A. Drozd, Y. Guo, T. Liu, S. Matsuoka, and X. Du, "Scaling word2vec on big corpus," *Data Science and Engineering*, vol. 4, pp. 157–175, 2019.

[68] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014.

[69] R. Shinde and A. H. Victoria, "Bert: A paradigm shift in natural language processing," in *Big Data Analytics and Data Science*, Lecture Notes in Networks and Systems, pp. 337–351, Springer, 2024.

[70] I. Gabsi, H. Kammoun, A. Wederni, and I. Amous, "Biobert for multiple knowledge-based question expansion and biomedical extractive question answering," in *Computational Collective Intelligence*, Lecture Notes in Computer Science, pp. 199–210, Springer, 2024.

[71] E. Sayers, "The ncbi e-utilities," *Nucleic Acids Research*, vol. 41, no. D1, pp. D408–D412, 2013.

[72] R. K. Mitchell, K. I. Berns, and D. J. Lipman, "Pubmed central: an archive of life sciences journals," *Nucleic Acids Research*, vol. 33, no. $\text{suppl}_1, pp.\ D223 - -D227, 2005$.

[73] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Kuttler, M. Lewis, W. tau Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *arXiv preprint arXiv:2009.07118*, 2020.

[74] A. Tucker, T. Kannampallil, S. J. Fodeh, and M. Peleg, "New jbi policy emphasizes clinically-meaningful novel machine learning methods," *Journal of Biomedical Informatics*, vol. 127, p. 104003, 2022.

[75] "Reglamento general de protección de datos (rgpd)." https://eur-lex.europa.eu/legal-content/ES/TXT/?uri=CELEX%3A32016R0679, 2016. Acceso el 17 de junio de 2025.

[76] "Health insurance portability and accountability act (hipaa)." https://www.hhs.gov/programs/hipaa/index.html, 1996. Acceso el 17 de junio de 2025.

[77] Organización Mundial de la Salud, "Estrategia mundial sobre salud digital 2020–2025." https://iris.who.int/bitstream/handle/10665/344251/9789240027572-spa.pdf, 2021. Consultado el 17 de junio de 2025.

Anexos

A Aspectos éticos, económicos, sociales y ambientales

"La tecnología es un sirviente útil pero un amo peligroso."
— Christian Lous Lange (1869-1938),

A.1 Introducción

Este Trabajo de Fin de Grado se centra en explorar cómo los modelos de lenguaje de gran tamaño (LLM) y las arquitecturas basadas en Retrieval-Augmented Generation (RAG) pueden contribuir al diagnóstico médico, facilitando una medicina más personalizada y basada en evidencia.

Esta línea de investigación responde a la creciente necesidad de mejorar la precisión, transparencia y accesibilidad de los sistemas de apoyo al diagnóstico, en un contexto donde la información médica se multiplica exponencialmente y el personal sanitario enfrenta limitaciones de tiempo y recursos.

Desde una perspectiva ambiental y económica, los LLM requieren una enorme cantidad de recursos computacionales tanto para su entrenamiento como para su aplicación práctica, lo que implica un elevado consumo energético y un costo asociado considerable.

Desde un punto de vista ético y social, el uso de modelos de lenguaje en contextos sanitarios implica riesgos relacionados con la fiabilidad de la información y la transparencia de las respuestas.

Este trabajo pone especial atención en evaluar estas cuestiones, valorando no solo el rendimiento técnico de los modelos, sino también su impacto en la calidad de la atención médica, con el objetivo de avanzar hacia una inteligencia artificial médica ética, sostenible y socialmente responsable.

A.2 Descripción de impactos relevantes relacionados con el proyecto

El uso de LLMs y arquitecturas RAG en el ámbito médico conlleva una serie de impactos relevantes.

Uno de los aspectos más relevantes es el alto consumo energético que requieren estos modelos, especialmente durante su uso en tiempo real. Aunque el entrenamiento es costoso, el uso continuo en aplicaciones como agentes conversacionales también supone un impacto ambiental significativo.

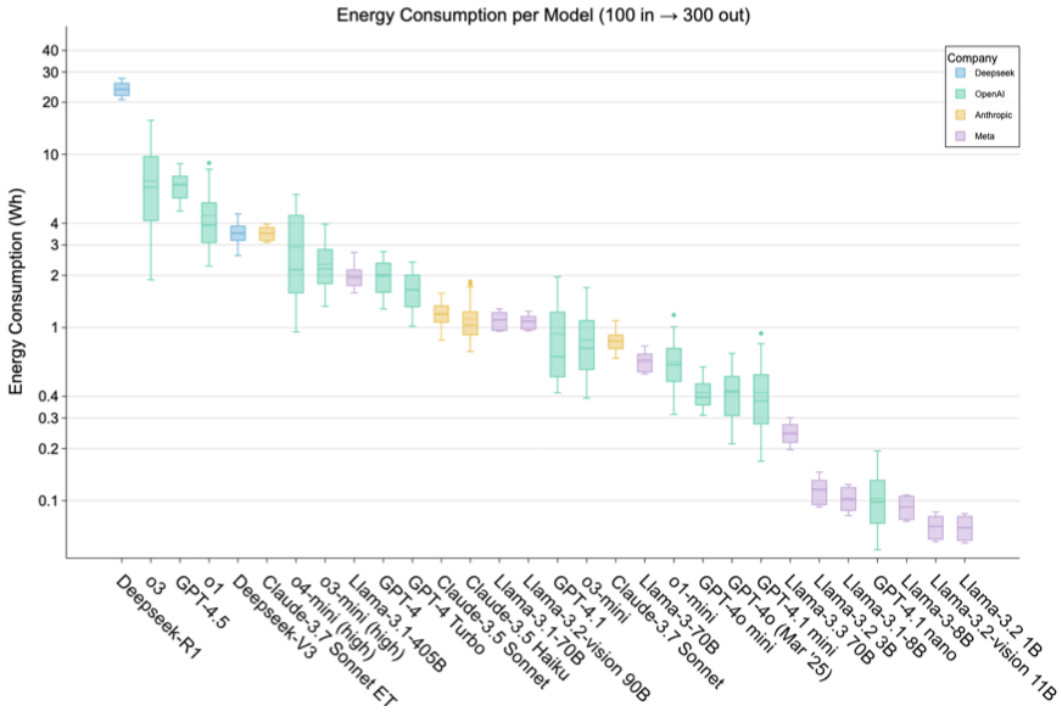


Figura 6.1: Consumo energético por consulta de inferencia en distintos LLMs [2].

Este proyecto ha empleado una infraestructura compartida basada en JupyterHub con GPU, lo que permite optimizar recursos respecto a soluciones comerciales, aunque sin eliminar totalmente la huella energética derivada del uso de hardware especializado.

Desde una perspectiva económica, cada consulta implica un coste operativo directo asociado al consumo energético y uso de GPU, que en entornos de nube se traduce en costes por token procesado o por tiempo de ejecución.

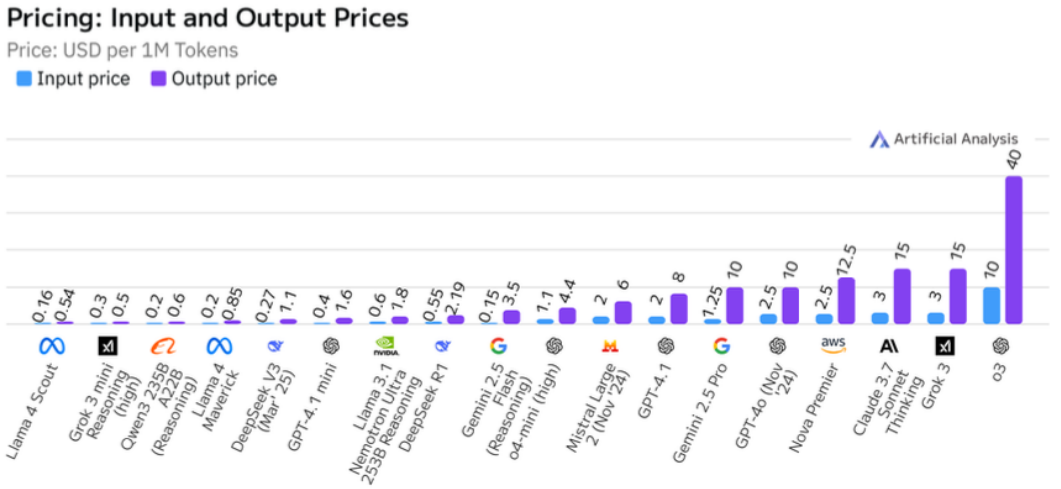


Figura 6.2: Comparativa de precios por millón de tokens de distintos LLMs [3]

Los LLM suelen estar entrenados principalmente en inglés, lo que puede afectar a su rendimiento en otros idiomas o culturas. Aunque no se ha realizado una evaluación multilingüe exhaustiva, se reconoce la importancia de mitigar sesgos lingüísticos y culturales en futuras implementaciones de sistemas RAG médicos.

Por último, se ha procurado que las respuestas generadas sean justificadas y basadas en evidencia médica actualizada, para evitar errores y preservar la confianza de médicos y pacientes. Este proyecto se ha diseñado para garantizar la transparencia en las respuestas y justificar cada diagnóstico con referencias médicas fiables, lo que representa un paso hacia una inteligencia artificial médica más segura y responsable.

A.3 Análisis detallado de alguno de los principales impactos

La adopción masiva de LLM está generando crecientes preocupaciones en torno a su impacto ambiental, debido principalmente al consumo energético sostenido que requieren para ofrecer servicios en tiempo real.

A pesar de que el entrenamiento inicial de estos modelos implica una elevada demanda energética y de recursos, diversos estudios han demostrado que su uso continuo puede superar, en volumen total de consumo, al propio entrenamiento, especialmente en aplicaciones como agentes conversacionales, donde el volumen de consultas diarias es potencialmente masivo [61].

Una sesión de 10 prompts consume unos 0.15 litros de agua, y una de 50, hasta 0.75 litros. Aunque el uso diario es bajo, entrenar modelos como GPT-3 puede requerir hasta 5.4 millones de litros, incluyendo 700,000 litros de consumo directo. Estos modelos operan en centros de datos que demandan gran energía y refrigeración, generando una notable huella de carbono [62].

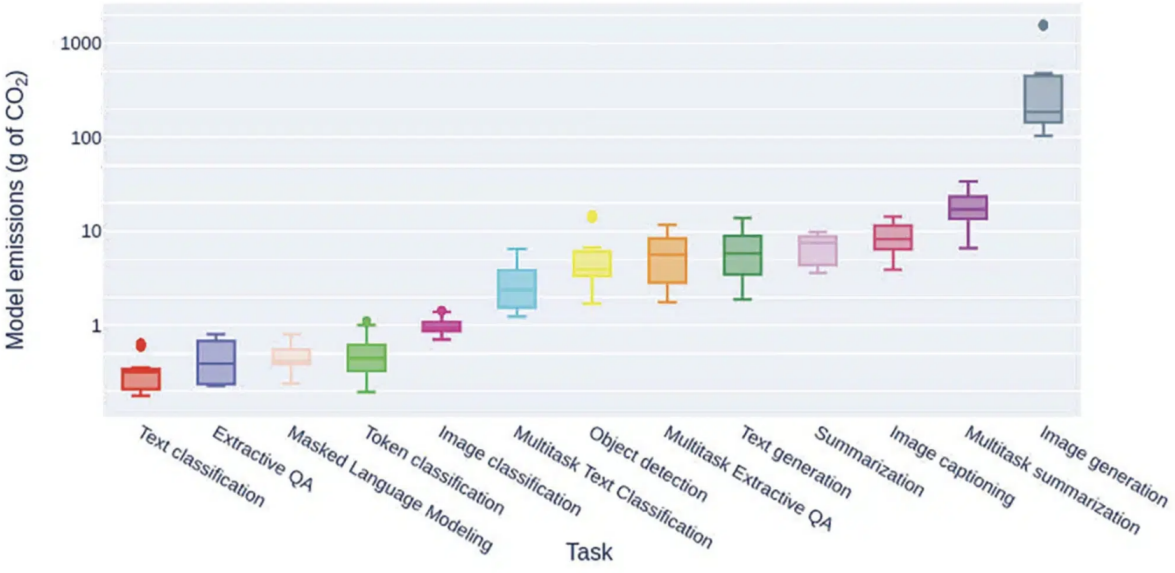


Figura 6.3: Distribución de emisiones de CO_2 por tarea [4]

Según estimaciones recientes, responder a una sola solicitud en un modelo tipo GPT-4 puede generar hasta 4 gramos de CO_2 equivalente (CO_2 eq), lo que multiplica por más de veinte la huella de una consulta web tradicional. Este impacto se ve amplificado por el crecimiento previsto del uso de la IA generativa en los próximos años [63].

La optimización energética y de recursos computacionales no solo tiene implicaciones medioambientales, sino que puede favorecer la implementación de modelos de IA médica en centros de salud que no disponen de grandes infraestructuras tecnológicas (hospitales con presupuestos limitados o países en vías de desarrollo).

Además, el diseño de sistemas más eficientes permite reducir la carga de los centros de datos y los efectos indirectos como la alteración de ecosistemas o la contaminación hídrica (agua) causada por los sistemas de enfriamiento industrial.

Por ello, es crucial concienciar sobre el impacto medioambiental de los modelos de inteligencia artificial y avanzar hacia soluciones más sostenibles. Este trabajo contribuye a ello, al utilizar modelos ligeros.

A.4 Conclusiones

Este trabajo se sitúa en un contexto donde influyen desafíos médicos y de sostenibilidad. La elección de técnicas como el fine-tuning de modelos preentrenados y la integración de RAG para reducir la frecuencia de entrenamientos intensivos contribuye a disminuir el consumo energético durante el desarrollo del agente conversacional para el diagnóstico clínico.

A la hora de evaluar el impacto económico, reutilizar y adaptar modelos existentes evita costos elevados de entrenamiento continuo. Esto promueve un acceso más amplio y equitativo, evitando que solo los grandes hospitales con amplios recursos puedan beneficiarse de asistentes conversacionales avanzados.

Estas mejoras técnicas deben ir acompañadas de criterios éticos, garantizando la privacidad de los datos, la transparencia del sistema y la accesibilidad lingüística y cultural para fomentar la confianza.

Aplicar principios de sostenibilidad, eficiencia y responsabilidad social no solo mejora el rendimiento, sino que también guía el desarrollo hacia una inteligencia artificial más respetuosa con el medio ambiente.

B Presupuesto económico

Para estimar los recursos utilizados durante el desarrollo de este Trabajo Fin de Grado, se ha elaborado el siguiente presupuesto económico.

COSTE DE MANO DE OBRA (coste directo)		**Horas**	**Precio/hora**	**Total**
		480	10 €	**4.800 €**
COSTE DE RECURSOS MATERIALES (coste directo)	**Precio de compra**	**Uso en meses**	**Amortización (en años)**	**Total**
Ordenador personal. Macbook pro M1 2020	1.679,00 €	8	10	**125,93 €**
JupyterHub con una GPU NVIDIA A100 40GB (proveedor Lambda Labs)	1,10 $/hora	8 (480 horas)	1	**489,60 €**
COSTE TOTAL DE RECURSOS MATERIALES				**615,53 €**

GASTOS GENERALES (costes indirectos)	15 %	sobre CD	**812,33 €**
BENEFICIO INDUSTRIAL	6 %	sobre CD+CI	**373,67 €**

MATERIAL FUNGIBLE	
Impresión	**100,00 €**
Encuadernación	**300,00 €**

SUBTOTAL PRESUPUESTO		**7.001,53 €**
IVA APLICABLE	21 %	**1.470,32 €**
TOTAL PRESUPUESTO		**8.471,85 €**

Tabla 6.1: Presupuesto Económico Detallado del Proyecto

C Tokenización y embeddings: el lenguaje en forma de números

Para que un modelo de lenguaje pueda procesar texto, primero debe convertirlo en una forma numérica. Este proceso comienza con la tokenización, que asigna un número a cada palabra o fragmento. Luego, estos tokens se transforman en vectores mediante embeddings, que capturan el significado y contexto de cada término.

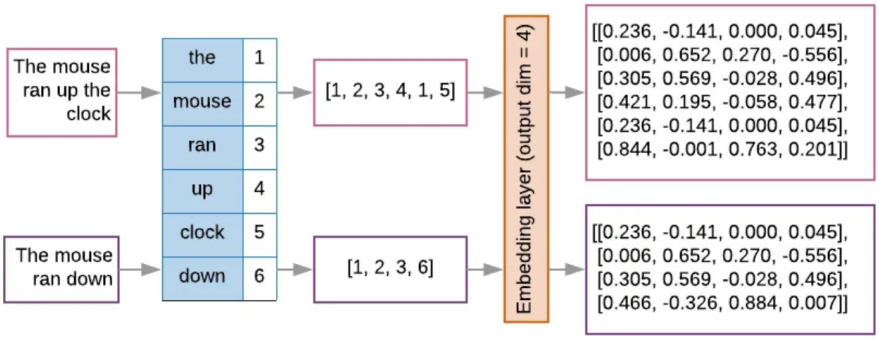


Figura 6.4: Ejemplo del proceso de tokenización y generación de embeddings [5]

C.1 Tokenizadores

En el contexto de los LLMs, un tokenizador es la herramienta que se encarga de dividir un texto en partes pequeñas llamadas tokens. Estos tokens pueden ser palabras completas, partes de palabras o incluso letras, dependiendo del idioma y del modelo empleado [64].

Una vez dividido el texto, cada token se convierte en un número que el modelo puede entender. Esta transformación es clave para que la IA pueda procesar el lenguaje humano.

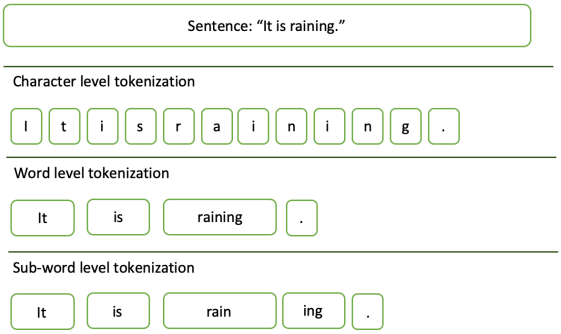


Figura 6.5: Ejemplo de fragmentación de texto por un tokenizador [6]

La forma en que se divide el texto en tokens influye directamente en el rendimiento del modelo. Si se generan muchos tokens pequeños, el procesamiento será más lento y costoso. Una tokenización eficiente agrupa fragmentos frecuentes en unidades más largas, lo que reduce la cantidad total de tokens y mejora tanto la velocidad como el uso de recursos.

Adaptar un tokenizador al lenguaje de los chatbots mejora su rendimiento. Las conversaciones suelen incluir jergas, abreviaciones, emojis y expresiones informales. Un tokenizador entrenado con datos reales de diálogo puede reconocer mejor estos elementos y representarlos con menos tokens.

Los estudios muestran que un vocabulario adaptado a conversaciones o contextos médicos puede reducir entre un 5 % y un 25 % el número de tokens necesarios. Esta reducción reduce el tiempo de procesamiento, ahorra energía y mejora la precisión y rapidez de las respuestas, al representar mejor el lenguaje médico [65].

En medicina, la eficiencia del tokenizador es clave, ya que los textos clínicos contienen términos técnicos y estructuras poco comunes que, si se fragmentan mal, afectan la precisión de los embeddings y la calidad de las respuestas del modelo. Un tokenizador especializado permite una compresión más eficiente del texto, manteniendo la calidad de la interacción mientras reduce los costos computacionales y energéticos.

C.2 Embeddings semánticos: cómo los modelos entienden el significado

Los modelos de lenguaje no pueden trabajar directamente con texto, por lo que necesitan convertirlo en números. Esta conversión se hace mediante los embeddings, que son vectores numéricos que representan el significado de palabras, frases o documentos [66].

En este espacio vectorial, los términos con significados similares están más cerca entre sí. Por ejemplo, "médico" y "doctor" tendrán vectores muy parecidos.

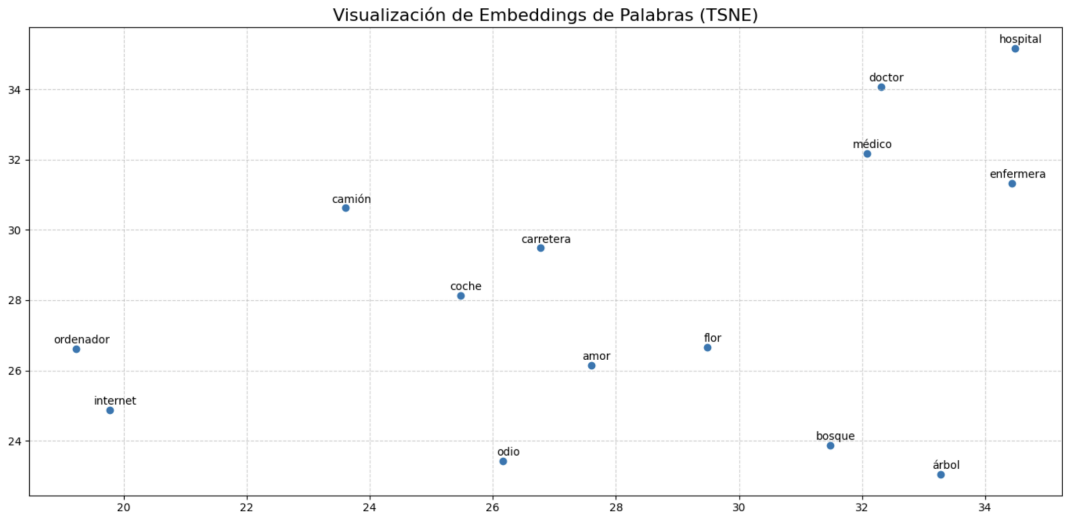


Figura 6.6: Visualización de embeddings de palabras mediante t-SNE

Existen dos tipos principales:

- Embeddings estáticos (Word2Vec [67], GloVe [68]): asignan un único vector por palabra, sin considerar el contexto, lo que puede generar ambigüedades semánticas.

- Embeddings contextuales (BERT [69], BioBERT [70]): adaptan el vector según el contexto, mejorando la precisión, especialmente en dominios como la medicina.

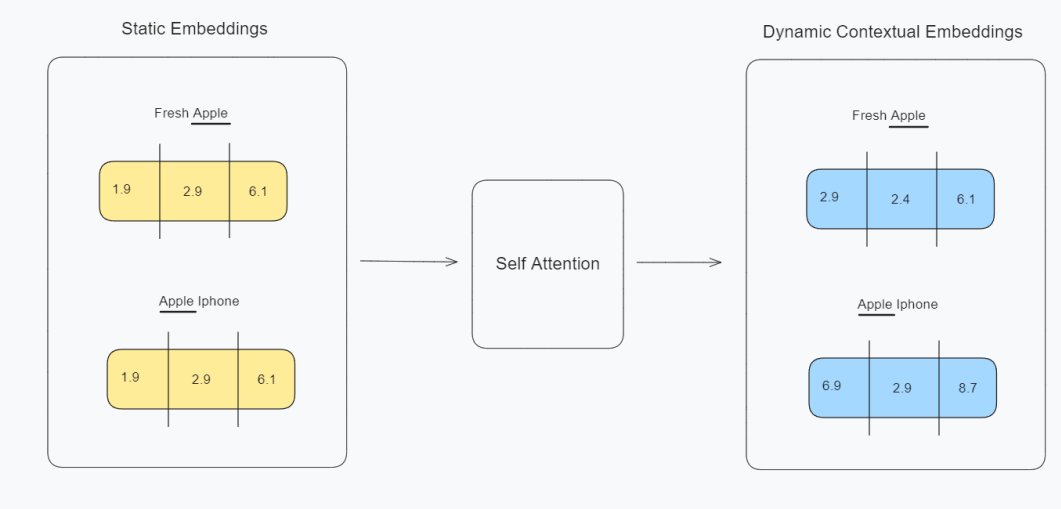


Figura 6.7: Comparación entre embeddings estáticos y contextuales [7]

En sistemas como RAG, los embeddings son esenciales. Primero se generan vectores para fragmentos de texto (como artículos médicos o definiciones). Luego, cuando el usuario hace una consulta, esta también se convierte en un embedding. El sistema compara estos vectores para encontrar los fragmentos más relevantes, permitiendo recuperar información útil basada en el significado, no solo en coincidencias literales (Figura 2.11).

Este tipo de búsqueda semántica mejora la precisión de las respuestas, ya que permite al modelo recuperar información útil aunque esté expresada de forma diferente.

C.3 Acceso a Información Científica mediante Entrez y PubMed

Una de las principales fuentes de conocimiento biomédico actual es la base de datos PubMed, accesible a través de las Entrez Programming Utilities (E-utilities) [71], proporcionadas por la National Library of Medicine (NLM) de Estados Unidos.

El agente conversacional utiliza esta API para realizar búsquedas semánticas automatizadas de artículos científicos relevantes [72].

El proceso se estructura en los siguientes pasos:

- Se lanza una consulta textual al servicio `esearch`, que devuelve una lista de identificadores (PubMed IDs) de los artículos más relevantes.
- A partir de estos IDs, se recupera información adicional mediante:
  - `esummary`, que proporciona títulos y descripciones breves.
  - `efetch`, que permite obtener el resumen completo (abstract) de cada artículo.
- Se filtran resultados vacíos o irrelevantes y se respeta un intervalo entre peticiones para cumplir con las políticas de uso de la API.

Los textos recuperados se combinan con fragmentos relevantes de una base de conocimiento local, previamente embebida mediante un modelo biomédico especializado. La similitud semántica entre la consulta del usuario y cada fragmento se calcula utilizando embeddings, seleccionando los más relevantes.

El contexto final, compuesto por información científica actualizada y conocimiento local, se organiza en un único prompt que se envía al modelo de lenguaje. Esto permite generar respuestas precisas, contextualizadas y basadas en evidencia.

Este enfoque, basado en la arquitectura RAG, permite mantener el sistema actualizado con la literatura médica más reciente sin necesidad de reentrenar el modelo base, lo que mejora su robustez y fiabilidad en entornos clínicos [73].

D Despliegue clínico: Validación, privacidad y requisitos técnicos

En este anexo se abordan reflexiones complementarias acerca del potencial de aplicación real del sistema conversacional médico propuesto.

D.1 Pruebas piloto para evaluar la utilidad práctica del sistema

Antes de implementar el agente conversacional en entornos clínicos reales, es fundamental validar su utilidad mediante pruebas piloto. Estas pruebas permiten evaluar no solo la precisión técnica del sistema, sino también su utilidad para los profesionales sanitarios y su integración en el flujo de trabajo clínico [74].

Una primera fase consiste en simulaciones de consultas médicas, donde los médicos prueban el sistema con ejemplos de pacientes cuyo diagnóstico ya se conoce, para comprobar si el agente llega a conclusiones similares.

El sistema ofrece sugerencias diagnósticas, pruebas complementarias o tratamientos, y los profesionales valoran la calidad y utilidad de las respuestas. Esto permite recoger tanto métricas objetivas (precisión, tiempo de respuesta) como valoraciones subjetivas (claridad, utilidad percibida).

Tras validar el sistema mediante simulaciones y pruebas controladas, se inicia una segunda fase centrada en su implementación en un entorno clínico real a lo largo de un período prolongado.

Durante esta etapa, se evalúa el impacto del agente conversacional, como la reducción de errores diagnósticos y los tiempos de atención. También se analiza su efecto en la satisfacción y la carga de trabajo del personal sanitario.

Esta evaluación permite no solo medir la precisión del modelo, sino también validar o rechazar el uso del asistente conversacional dentro del entorno sanitario.

Además, es importante medir la experiencia de usuario mediante encuestas y entrevistas a los profesionales participantes, valorando aspectos como la facilidad de uso, la claridad del lenguaje, la rapidez del sistema y la atractividad del entorno web.

Dado que estas pruebas pueden contener información médica, es fundamental cumplir con la ley de protección de datos y la participación voluntaria explicita y escrita del paciente.

Estas pruebas piloto son clave para validar el sistema en condiciones reales y asegurar que su integración en el entorno sanitario sea útil, segura y ética.

D.2 Gestión de la privacidad del paciente en entornos hospitalarios

La integración de un agente conversacional en hospitales exige especial atención a la privacidad y seguridad de los datos clínicos. Al manejar información sensible como síntomas o historiales médicos, el sistema debe cumplir estrictamente con normativas como el RGPD en Europa [75]

y la HIPAA en EE. UU. [76].

Se recomienda que el agente funcione en servidores locales del hospital o en infraestructuras privadas certificadas, evitando el envío de datos a servicios externos. Además, toda la información debe transmitirse cifrada y con autenticación segura.

Es fundamental definir distintos niveles de acceso según el rol de cada usuario y mantener un registro detallado de todas las interacciones con el sistema. Esto permite detectar accesos no autorizados, errores o usos indebidos, y garantiza un control seguro sobre la información almacenada.

Es obligatorio obtener el consentimiento informado del paciente, explicando de forma clara cómo se utilizarán sus datos, con qué finalidad y durante cuánto tiempo.

En resumen, garantizar la privacidad no es solo una obligación legal, sino una condición esencial para una implementación ética y segura de la inteligencia artificial en el ámbito sanitario.

D.3 Requisitos técnicos para su despliegue en hospitales

Elegir cómo implementar un agente conversacional en un hospital es una decisión compleja. Es necesario analizar con cuidado factores clave como la seguridad de los datos, la compatibilidad con los sistemas clínicos existentes y los recursos técnicos disponibles [77].

Una opción es el despliegue en la nube privada, que permite una gestión centralizada, actualizaciones más ágiles y una mayor escalabilidad. Esta modalidad es ideal para centros que buscan eficiencia operativa y flexibilidad, aunque implica una mayor dependencia de la conectividad y requiere garantizar que los datos estén protegidos bajo normativas estrictas.

La otra alternativa es la instalación local dentro del propio hospital. Esta opción ofrece un mayor control sobre los datos y no depende de la conexión a internet, lo que puede ser clave en entornos con requisitos de seguridad muy estrictos. Sin embargo, supone una mayor inversión en infraestructura y mantenimiento técnico.

Independientemente del enfoque elegido, el sistema debe cumplir con los estándares de interoperabilidad y protección de datos, y adaptarse a las necesidades operativas del hospital. Solo así podrá garantizarse una integración segura, sostenible y aceptada por los profesionales sanitarios.