TRABAJOS FIN DE GRADO

Universidad de Jaén

Escuela Politécnica
Superior de Jaén

BoTCA: Generación de Lenguaje Confiable para la Prevención de los Trastornos de la Conducta Alimentaria

Autor: María Teresa Muñoz Martín

Grado en Ingeniería Informática

Director: Prof. D. Eugenio Martínez Cámara
Departamento del director: Departamento de Informática

Fecha: 30/05/2024

AGRADECIMIENTOS

Me gustaría agradecer a todas las personas que han estado a mi lado apoyándome en esta etapa. A mi familia y amigos, en especial, a mis padres y mi hermana, que me han acompañado y ayudado en todo momento. Gracias por darme lo mejor de vosotros en cada momento y enseñarme a bailar bajo la lluvia. Sin duda, no sería la persona que soy hoy sin vosotros a mi lado.

Por otra parte, quiero agradecer a los profesores que han formado parte de mi vida académica, especialmente a mi tutor, Eugenio Martínez Cámara, por confiar en mi y guiarme en todo lo posible. También, agradecer al grupo SINAI por acogerme y ayudarme en el último empujón de la carrera. A todos los compañeros que he conocido y con los que he podido compartir esta etapa de mi vida.

Ficha del Trabajo Fin de Título

Titulación	Grado en Ingeniería Informática
Modalidad	Proyecto de Ingeniería
Especialidad (solo TFG)	Sistemas de Información
Mención (solo TFG)	
Idioma	Español
Tipo	Específico
TFT en equipo	No
Autor/a	María Teresa Muñoz Martín
Fecha de asignación	16/04/2024
Descripción corta	Este trabajo presenta el desarrollo de un sistema de generación de lenguaje confiable para la prevención de los Trastornos de la Conducta Alimentaria haciendo uso de herramientas de aprendizaje profundo y técnicas de Procesamiento del Lenguaje Natural

Índice

Capítulo 1

Introducción

1.1. Motivación

La salud mental es un estado de equilibrio y bienestar psicológico que permite a las personas afrontar situaciones de estrés, desarrollar sus habilidades, aprender, trabajar de forma adecuada y contribuir positivamente a su comunidad. Es un aspecto esencial de nuestra salud y bienestar general, que influye en nuestra capacidad para tomar decisiones, establecer relaciones y dar forma a nuestro entorno. También es un derecho humano fundamental y vital para el crecimiento personal, comunitario y socioeconómico [1].

A lo largo de nuestra vida, diversos factores individuales, sociales y estructurales pueden influir en nuestra salud mental, ya sea protegiéndola o afectándola negativamente. Entre estos factores se encuentran aspectos psicológicos y biológicos, como las habilidades emocionales, el consumo de sustancias y factores genéticos, que pueden aumentar la vulnerabilidad de una persona a trastornos de salud mental. Es importante destacar que la salud mental va más allá de la mera ausencia de trastornos mentales. Se trata de un proceso complejo que varía en cada individuo, con distintos niveles de dificultad y angustia, así como resultados sociales y clínicos diversos.

En los últimos años, la salud mental ha ganado mayor visibilidad y aceptación, superando los tabúes y prejuicios que solían rodearla injustamente. Este cambio es fundamental, ya que la salud mental es un tema de gran importancia debido a su impacto en el bienestar personal, como lo reconoce la Organización Mundial de la Salud (OMS)[1].

Por otra parte. la pandemia de COVID-19 ha tenido un impacto significativo en la salud mental de millones de personas en todo el mundo. Según el Centro de Investigaciones

[1] https://www.who.int/es/health-topics/mental-health

Sociológicas (CIS), un 6,4 % de la población ha buscado ayuda profesional para síntomas relacionados con la salud mental desde el comienzo de la pandemia. De estas consultas, el 43,7 % se debió a ansiedad y el 35,5 % a depresión. Además, se observa que más del doble de mujeres que hombres han buscado estos servicios de salud mental [2].

Aunque la depresión y la ansiedad son las enfermedades mentales más frecuentes [2], desgraciadamente no son las únicas que afectan a la sociedad actual. Recientes estudios y noticias han resaltado la creciente prevalencia y gravedad de los TCA en la sociedad moderna [3]. Estos trastornos pueden llevar a complicaciones severas y crónicas, y en casos extremos, pueden resultar en la muerte. Además, estos afectan a un amplio espectro de la población, incluyendo a niños, adolescentes y adultos, y no discriminan por género, aunque hay una prevalencia notablemente alta en mujeres jóvenes. La visibilidad y sensibilización sobre los TCA es crucial para fomentar una comprensión adecuada, prevención y tratamiento efectivo. Por estas razones, la atención a los TCA no solo mejora la calidad de vida de los afectados, sino que también ayuda a reducir la incidencia de otros trastornos mentales relacionados, como la depresión y la ansiedad. Este enfoque integrado puede llevar a una mejor salud mental y física a largo plazo para quienes padecen estas condiciones. Es por eso que en este trabajo, nos vamos a centrar en los Trastornos de la Conducta Alimentaria.

Los TCA son trastornos que involucran comportamientos persistentemente alterados relacionados con la alimentación que suelen comenzar en la adolescencia o al inicio de la adultez, causando graves repercusiones tanto físicas como psicológicas. Aunque los TCA más comunes son la Anorexia Nerviosa (AN), la Bulimia Nerviosa (BN) y el trastorno por atracón, en las clasificaciones actuales de los trastornos mentales, como el DSM-5 [3], también se incluyen otros trastornos alimentarios como el trastorno por evitación/restricción de la ingesta, especialmente vistos en la infancia.

Al igual que con otras enfermedades, la pandemia trajo consigo un aumento de casos graves de TCA. Los psiquiatras lo achacan a diferentes motivos, incluido el hecho de que pasar más tiempo con la familia, ha implicado que haya una mayor detección [4]. Sin embargo, el factor que mayor impacto ha tenido parece ser que se centra en el aumento del uso de las Redes Sociales (RRSS). Con el confinamiento las personas pasaban más tiempo con las redes, siguiendo a gente para ver cómo alimentarse adecuadamente o cómo hacer ejercicio. Todo esto ha aumentado la obsesión aún más, agravando la situación de estos trastornos.

[2] La depresión afecta a cerca de 264 millones de personas [2].

[3] `https://www.eafit.edu.co/ninos/reddelaspreguntas/Documents/dsm-v-guia-consulta-manual-diagnostico-estadistico-trastornos-mentales.pdf`

Además, debemos destacar que la mayoría de la sociedad subestima la gravedad de los trastornos de la alimentación, considerándolos en muchas ocasiones como una moda o incluso como un estilo de vida saludable. Como resultado, las familias suelen carecer de conciencia sobre el problema hasta que la enfermedad está en una etapa avanzada, lo que dificulta su detección temprana y agrava la situación. Es por esto que hemos decidido centrarnos en los TCA para nuestro trabajo. Para ello, aplicamos técnicas de Procesamiento del Lenguaje Natural (PLN), y planteamos distintas propuestas, hasta llegar a la decisión de implementar un chatbot para intentar generar respuestas a las necesidades que tengan aquellas personas que padecen estos trastornos, o cualquiera que quiera entender más sobre ellos, como familiares o amigos de personas afectadas.

1.2. Propósito

En primer lugar, este trabajo consiste en el estudio de distintas alternativas que existen para poder ayudar a las personas que sufren un TCA. Hay que destacar, que no se pretende realizar un análisis o detección de la enfermedad como tal, sino que nuestros objetivo es ayudar a personas que quieran aprender sobre estos trastornos o estén en un momento de estrés y ansiedad, puedan acudir a un sistema que esté disponible en cualquier momento con el fin de obtener información veraz y confiable.

Nuestro enfoque se centra específicamente en la experimentación con diversas estrategias destinadas a mejorar los resultados alcanzados por los actuales modelos del lenguaje de gran escala (LLM). Concretamente, lo que queremos conseguir con este trabajo es:

1. **Realización de un chatbot informativo para familiares, amigos y todo aquel que quiera informarse sobre estos trastornos:** Proporcionar una herramienta accesible y fácil de usar que pueda responder preguntas y brindar apoyo a quienes buscan información sobre TCA, ayudando a eliminar mitos y proporcionar datos precisos y actualizados.

2. **Generación de información confiable sobre TCA:** Garantizar que la información proporcionada sea rigurosa y basada en evidencias científicas, contribuyendo así a la educación y la concienciación sobre estos trastornos.

3. **Creación de un sistema que esté disponible en cualquier momento:** Desarrollar un chatbot que funcione 24/7, ofreciendo asistencia inmediata y continua, sin importar la hora o el lugar, para brindar apoyo en momentos de necesidad.

4. **Fomentar la detección temprana de síntomas:** Ayudar a usuarios a identificar signos tempranos de TCA a través de información y preguntas orientativas, promoviendo la búsqueda de ayuda profesional oportuna.
5. **Reducir la estigmatización asociada con los TCA:** Utilizar el chatbot como una herramienta educativa que ayude a normalizar la conversación sobre los TCA, reduciendo el estigma y alentando a las personas afectadas a buscar ayuda.
6. **Adaptabilidad y personalización de respuestas:** Implementar algoritmos que permitan personalizar las respuestas del chatbot según las necesidades y el contexto del usuario, haciendo la interacción más relevante y efectiva.
7. **Evaluación continua y mejora del chatbot:** Establecer un sistema de retroalimentación y análisis de datos para evaluar la efectividad del chatbot y realizar mejoras continuas basadas en la experiencia del usuario y los avances en la tecnología de LLM.

1.3. Objetivos

El objetivo principal que se desea conseguir con este TFG consiste en desarrollar un agente conversacional que sea capaz de responder de manera veraz y confiable a preguntas relacionadas con los TCA. Para conseguir este objetivo global debemos centrarnos en desarrollar una serie de subojetivos que se detallan a continuación:

- Estudiar el estado de la cuestión para conocer qué técnicas son las más adecuadas para abordar el problema de los TCA.
- Conocer las necesidades de las personas que sufren estas enfermedades para saber cómo enfrentarnos a ellos.
- Investigar sobre los LLMs existentes y cuáles son los que generan las respuestas más apropiadas para esta tarea.
- Buscar los documentos que se usarán como base de conocimiento para que el sistema pueda contestar a las consultas propuestas.
- Desarrollar un sistema que sea capaz de generar respuestas en base a las necesidades de los usuarios, usando el corpus seleccionado anteriormente.
- Analizar los resultados de distintas alternativas, seleccionando el que mejor se adapte a las necesidades de las personas que sufren TCA.

- Redactar una memoria que recoja la información del trabajo desarrollado.

1.3.1. Objetivos de Desarrollo Sostenible

Por último, me gustaría resaltar que este TFG se encuentra alineado con los Objetivo de Desarrollo Sostenible (ODS) para la agenda 2030[4]. El 25 de septiembre de 2015 los líderes mundiales adoptaron 17 ODS para proteger el planeta, luchar contra la pobreza y tratar de erradicarla con el objetivo de construir un mundo más próspero, justo y sostenible para las generaciones futuras. Estos objetivos se fijaron dentro de la Agenda 2030 sobre el desarrollo sostenible. Concretamente este TFG se alinea con el **ODS número 3: Salud y bienestar**. Dicho ODS tiene como objetivo fundamental garantizar una vida saludable y promover el bienestar para todas las personas en todas las etapas de la vida. Esto implica asegurar el acceso universal a servicios de salud de calidad, incluyendo la prevención, el tratamiento y la atención de enfermedades, así como promover prácticas saludables y estilos de vida activos. Además, busca reducir la mortalidad materna, neonatal e infantil, así como abordar las enfermedades transmisibles y no transmisibles. **El ODS 3 también se centra en mejorar la salud mental y el bienestar emocional, así como garantizar la cobertura sanitaria universal para que nadie se quede atrás en el acceso a los servicios de salud esenciales.** Es por ello que el desarrollo de este proyecto impacta de manera directa en la consecución del ODS 3, si bien, también afecta a otros ODS como son:

1. **ODS 4. Educación de calidad:** Busca garantizar una educación inclusiva, equitativa y de calidad, y promover oportunidades de aprendizaje durante toda la vida para todos. Nuestro trabajo se alinea con este objetivo al educar sobre los TCA, aumentar la conciencia pública y promover la comprensión de estos trastornos.

2. **ODS 5. Igualdad de género:** Se centra en lograr la igualdad de género y empoderar a todas las mujeres y niñas. Los TCA afectan desproporcionadamente a mujeres y niñas, por lo que abordar estos trastornos puede contribuir a reducir las disparidades de género en salud mental.

3. **ODS 10. Reducción de desigualdades:** Busca reducir las desigualdades dentro y entre los países. Los TCA pueden afectar a personas de diferentes orígenes socioeconómicos, pero las barreras de acceso a la atención médica y los recursos pueden exacerbar estas disparidades. Este proyecto mejora el acceso a la atención y apoyo para todos.

[4] `https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/`

1.4. Alcance

Una vez establecidos los objetivos generales y requisitos específicos del proyecto, se espera obtener los siguientes entregables y resultados al finalizar:

- **Código fuente:** Se proporcionará el código fuente utilizado en todos los experimentos realizados durante el proyecto.
- **Resultados:** Además del código, se presentarán los resultados obtenidos en cada experimento, detallando las conclusiones alcanzadas.
- **Memoria:** Este documento actúa como la memoria del proyecto, que incluye toda la información relacionada con el desarrollo del proyecto, detallando todo lo relacionado con los resultados de los experimentos.
- **Vídeo:** Se incluirá un vídeo con una demostración visual mostrando la ejecución del sistema.

1.5. Planificación temporal

El propósito de la planificación de un proyecto es definir un alcance claro y delimitado que facilite la realización de estimaciones precisas sobre el costo y la duración del proyecto. Estas estimaciones se han ido ajustando conforme avanzaba el desarrollo del proyecto.

En la Tabla 1.1, podemos ver cada fase para entender mejor esta planificación:

Fase	Fecha de inicio	Objetivo
Fase 1	1/02/2024	Investigación inicial sobre las enfermedades mentales, en particular, los trastornos de la conducta alimentaria.
Fase 2	12/02/2024	Análisis sobre las distintas técnicas de PLN existentes para poder aplicarlas a nuestro proyecto.
Fase 3	19/02/2024	Recopilación de datos para nuestro estudio sobre los TCA.
Fase 4	26/02/2024	Fine-tuning de nuestros modelos y realización de pruebas con esta técnica.
Fase 5	4/03/2024	Estudio de alternativas al fine-tuning.
Fase 6	11/03/2024	Desarrollo de nuestro chatbot con la técnica de RAG.
Fase 7	1/04/2024	Análisis de resultados.
Fase 8	15/04/2024	Elaboración de la memoria.

Tabla 1.1: Planificación del proyecto

En la Figura 1.1 se muestra un diagrama de Gantt, donde se visualiza de manera más clara, el inicio y fin de cada una de las fases de nuestro proyecto.

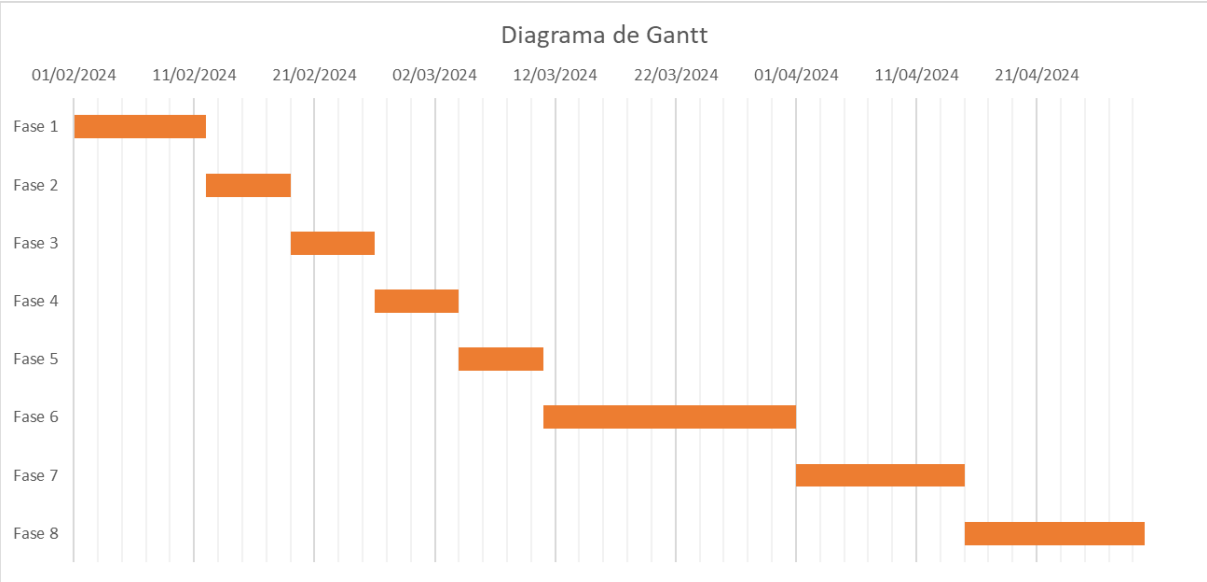

Figura 1.1: Diagrama de Gantt

1.6. Presupuesto

En este apartado presentamos una estimación del coste del trabajo teniendo en cuenta las distintas tecnologías que vamos a usar, las tareas a realizar, el personal para desarrollar estas tareas y los servicios contratados. Para calcular el coste total, hemos tenido en cuenta el tiempo de duración del proyecto, que hemos calculado que serán 44 días laborales, suponiendo aproximadamente dos meses de trabajo.

Para entenderlo de manera más clara, se muestran en los siguientes apartados los costes para los productos software, hardware y de contratación del personal necesarios para este proyecto.

1.6.1. Presupuesto software

Los costes software son aquellos derivados a los productos o herramientas software usados en este trabajo. Así podemos ver que todo el software utilizado en este proyecto es gratuito, excepto el sistema operativo. Como se trata de un bien amortizable, debemos calcular su amortización. En la Tabla 1.2 podemos ver los gastos.

Software	Coste	Tiempo de Vida Útil	Coste Final	Total
Windows 11 Home	145 euros	2 años	6,04 euros/mes	12,08 euros

Tabla 1.2: Presupuesto Software

1.6.2. Presupuesto hardware

Los costes hardware hacen referencia a la parte física del sistema informático. Así pues, en la Tabla 1.3 podemos ver los costes correspondientes de hardware utilizado en este proyecto.

Hardware	Coste	Tiempo de Vida Útil	Coste Final	Total
HP Spectre x360 Convertible 13-aw0xxx	1.019,15 euros	4 años	21,23 euros/mes	42,46 euros

Tabla 1.3: Presupuesto Hardware

1.6.3. Presupuesto de personal

Por último, los costes de personal son los derivados de los profesionales que van a desarrollar este proyecto y que dependerán de los trabajos y especialización que tengan, puesto que ello determinará el salario. La Tabla 1.4 muestra la distribución del gasto en personal en este trabajo.

	Sueldo mensual	Número de meses	Coste final
Ingeniero informático	2.500 euros	2 meses	5.000 euros

Tabla 1.4: Presupuesto de personal

1.6.4. Presupuesto total

Presupuesto Software	12,08 euros
Presupuesto Hardware	42,46 euros
Presupuesto Personal	5.000 euros
Presupuesto Total	5.054,54 euros

Tabla 1.5: Presupuesto Total

Capítulo 2

Antecedentes y estado de la cuestión

2.1. Trastornos de la Conducta Alimentaria

Como hemos explicado en la introducción del proyecto, los trastornos de la alimentación, también conocidos como TCA, son enfermedades mentales que se caracterizan por una conducta alterada ante la ingesta alimentaria y/o la aparición de comportamientos encaminados a controlar el peso [5]. Esta alteración provoca problemas físicos y del funcionamiento psicosocial. Los TCA son principalmente caracterizados por un desorden en los hábitos alimenticios y una obsesión extrema por la percepción de uno mismo y el peso corporal. Dentro de este conjunto de enfermedades, destacan la Anorexia Nerviosa (AN), la Bulimia Nerviosa (BN), los trastornos por atracón y otros no específicos. Las nuevas tendencias en la moda y los estándares en el aspecto físico y patrones de alimentación se señalan como posibles disparadores del aumento de dichos trastornos. La etiopatogenia de los TCA aún no se conoce aunque se sabe que es un trastorno de causa multifactorial. Dentro del debate científico sobre la causa de estos, se señalan distintos factores socioculturales como el énfasis en la delgadez, la presión de los roles de género, historias de abuso sexual y problemas familiares, entre otros.

El TCA es uno de los problemas de salud mental que más afecta en los últimos años a la población joven, especialmente a las mujeres [6]. Solo en Cataluña representan el 92,4 % de los casos registrados, según los últimos datos disponibles del Servicio Catalán de Salud [7]. Según la Asociación Contra la Anorexia y la Bulimia (ACAB) [8], en 9 de cada 10 casos la persona afectada es una mujer. Aunque se advierte que los TCA no entienden de género, pues actualmente, también ha aumentado la presión social hacia el cuerpo masculino y los casos van incrementando cada vez más y más. Aún así, sigue habiendo una diferencia muy significativa entre los cánones que genera la sociedad hacia el cuerpo femenino y masculino.

En lo que respecta a España, en general, los datos apuntan a que los TCA afectan a uno de cada 20 adolescentes, y la edad media de inicio está disminuyendo, colocándose actualmente en los 12 años y medio. Según la Sociedad Española de Médicos Generales y de Familia [9], estos trastornos tienen una prevalencia del 4,5 % entre las mujeres jóvenes de 12 a 21 años mientras que solo afecta al 0,3 % de los hombres. Además, se ha notado un incremento del 20 % en los ingresos hospitalarios en servicios especializados durante el último año, asociado en parte a los efectos de la pandemia en la salud mental.

Este problema no se limita a España; a nivel global, los casos de TCA se han duplicado en las últimas dos décadas. En el caso de la anorexia, ha llegado a ser la enfermedad mental con la tasa de mortalidad más alta, debido a complicaciones como enfermedades cardiovasculares, desequilibrios endocrinos y trastornos gastrointestinales, además del incremento de la tasa de suicidios entre los pacientes afectados.

Influencia de las RRSS

Las Redes Sociales (RRSS) son como un espejo para los jóvenes, reflejando estándares de belleza inalcanzables que pueden impactar negativamente en su autoestima y percepción de sí mismos. Éstas desempeñan un papel importante en el desarrollo y la persistencia de los TCA, actuando a menudo como un desencadenante al fomentar la preocupación por la imagen corporal y la alimentación. En aquellas personas que ya sufren de TCA, las RRSS pueden dificultar la conciencia de su enfermedad al normalizar comportamientos poco saludables y reforzar creencias negativas sobre sí mismos a través de comparaciones constantes con otros. Así pues, la influencia de las RRSS, la idealización de la delgadez, la insatisfacción corporal y la creencia errónea de que la delgadez conduce a la felicidad, pueden promover estos trastornos alimentarios.

Las redes han creado un entorno donde la comparación constante y la búsqueda de validación son moneda corriente, lo que puede tener efectos significativos en la salud mental de los adolescentes. La exposición a imágenes retocadas y cuerpos "perfectos" promovidos en plataformas como Instagram o TikTok puede generar una percepción distorsionada de la realidad y aumentar la insatisfacción corporal en los adolescentes. Esta insatisfacción puede ser un factor de riesgo para el desarrollo de estas enfermedades. Además, las interacciones en las redes sociales pueden influir en la forma en que los adolescentes perciben y manejan la alimentación. La exposición a contenido relacionado con dietas extremas, ejercicios intensos y consejos poco saludables de pérdida de peso puede fomentar conductas alimentarias desordenadas y distorsionar la relación de los adolescentes con la comida.

Por consiguiente, es esencial comprender que nuestros dispositivos móviles actúan como una ventana interminable de información. Aunque el uso de estos dispositivos tiene aspectos positivos, también conlleva riesgos, como la idealización de la apariencia física y la falta de aceptación de nuestros propios cuerpos debido a comparaciones con 'influencers' que admiramos.

Impacto de la COVID-19

A parte de las RRSS, otro factor que ha hecho aumentar estos trastornos ha sido la pandemia de COVID-19. Esta ha alterado profundamente la vida cotidiana en todo el mundo, con efectos negativos tanto en la salud física como mental. La situación ha impactado de manera significativa en toda la población, con un especial énfasis en la población juvenil, quienes han experimentado aislamiento social, preocupaciones por contagios y el cambio repentino a clases virtuales, lo que ha provocado un aumento en síntomas psicológicos como ansiedad, estrés, angustia, depresión o trastornos del sueño, todos los cuales representan un riesgo para el desarrollo de los TCA.

El cambio a un estilo de vida más sedentario y el acceso limitado a alimentos saludables debido a restricciones de movimiento y cierres de negocio pueden influir en los patrones alimentarios de las personas con TCA, agravando comportamientos restrictivos o compulsivos en torno a la comida. Además, la interrupción de los servicios de atención médica y tratamiento especializado debido a la pandemia dificultó la recuperación y el manejo de los TCA. Muchos programas de tratamiento han tenido que reducir sus servicios o cambiar a modalidades virtuales, lo que puede no ser adecuado para todos los pacientes, especialmente aquellos que requieren un apoyo más intensivo o supervisión médica. El aislamiento social impuesto por las medidas de distanciamiento aumentó los sentimientos de soledad en personas con TCA, lo que desencadenó comportamientos alimentarios desordenados como una forma de hacer frente a estas emociones difíciles.

Un estudio llevado a cabo en Hong Kong [10] examinó los cambios en la dieta, los síntomas asociados con los TCA, la ansiedad, la depresión y el bienestar psicológico. Descubrieron que el 86,1 % de los participantes, sin diagnóstico previo de TCA, realizaron al menos un cambio en su dieta durante este período. Además, el 46,2 % de la muestra describió su estilo de alimentación como poco saludable, lo que aumentó la probabilidad, hasta en un 25 %, de que estos individuos pudieran desarrollar un TCA, a pesar de no tener un diagnóstico previo de dicho trastorno. Este fenómeno generó un aumento generalizado en la ansiedad y la depresión debido al sentimiento de culpa asociado con los cambios en la imagen corporal, lo que a su

vez condujo a una insatisfacción con el propio cuerpo. Estos resultados destacaron el aumento en la probabilidad de desarrollar un TCA en aquellos individuos que habían modificado sus dietas, así como una mayor probabilidad de experimentar ansiedad o depresión. Siguiendo una línea similar, estudios realizados en hospitales de Barcelona han mostrado cómo, en tan solo las dos primeras semanas de confinamiento, los pacientes con TCA experimentaron preocupaciones vitales más intensas en áreas laborales, sociales y personales, entre otras; la gravedad de la sintomatología presentada aumentó en un 38 %; se observó un incremento del 56,2 % en los síntomas adicionales de ansiedad; y el estrés generado por todo esto potenció la alimentación emocional que ya estaban experimentando [11].

En conclusión, los trastornos alimenticios son enfermedades que están a la orden del día, siendo una enfermedad que cada vez tiene mayor impacto en la sociedad. Esto explica la motivación de este proyecto, mediante el cual, con herramientas de PLN, hemos querido aportar alguna solución para ayudar a esas personas que se ven afectadas.

2.2. Inteligencia Artificial y Procesamiento del Lenguaje Natural

Según la RAE, la Inteligencia Artificial (IA) es la "disciplina científica que se ocupa de crear programas informáticos que ejecutan operaciones comparables a las que realiza la mente humana, como el aprendizaje o el razonamiento lógico". Así pues, es un campo de la informática que se enfoca en investigar sistemas informáticos capaces de llevar a cabo tareas que normalmente requieren inteligencia humana, como el razonamiento y la conducta. Las primeras aplicaciones en este campo se centraron en desarrollar algoritmos para juegos. Hoy en día, abarca una variedad de campos dentro de la teoría de la computación. Esto incluye áreas como:

- **Aprendizaje automático:** está enfocada en crear algoritmos y modelos que pueden aprender por sí mismos a partir de los datos que se les proporciona.
- **Procesamiento del Lenguaje Natural (PLN):** se refiere a la capacidad de las máquinas para interpretar, interactuar y resolver problemas usando el lenguaje propio de un ser humano.
- **Visión por computador:** se centra en desarrollar algoritmos que permiten a las máquinas entender y trabajar con imágenes y vídeos.
- **Computación evolutiva:** se basa en principios de evolución y selección natural para buscar soluciones óptimas a problemas mediante simulación de procesos evolutivos.

- **Robótica:** es el estudio y desarrollo de sistemas que pueden operar de forma autónoma en entornos físicos.
- **Sistemas multiagentes:** implica la creación de programas que cooperan para alcanzar objetivos comunes.
- **Lógica difusa:** estudia el razonamiento más allá del binario, siendo útil en aplicaciones donde las condiciones no son simplemente verdadero o falso.

La definición más amplia y generalizada de la IA, establecida en el ámbito europeo, se refiere a la capacidad de una máquina para exhibir habilidades similares a las de los seres humanos, como el razonamiento, el aprendizaje, la creatividad y la capacidad de planificación [12]. En el desarrollo de este campo surgió el PLN, que ha estado en evolución desde la década de 1960. Este es una rama de la IA que se centra en desarrollar herramientas y técnicas para mejorar la comunicación entre humanos y máquinas utilizando el lenguaje natural. Este tipo de lenguaje, que utilizamos cotidianamente, es más flexible y menos estructurado que los lenguajes formales como los de programación o las fórmulas matemáticas. El objetivo principal es proporcionar a los sistemas automáticos la capacidad de comprender y generar el lenguaje como lo hacemos los humanos, utilizando diversos modelos y técnicas. Es un campo interdisciplinario que combina lingüística y aprendizaje automático con el objetivo de comprender todo lo relacionado con el lenguaje humano. Son muchas las tareas relacionadas con el PLN y que abordan diferentes desafíos [13]:

- **Clasificación automática de textos:** Consiste en agrupar automáticamente documentos, comentarios y otros textos en función del tema sobre el que tratan u otras características.
- **Anotación automática de textos:** Permite hacer un primer análisis lingüístico y etiquetado de cualquier texto, de forma que quede estructurado para aplicar distintas tareas de PLN.
- **Detección de entidades:** Permite localizar y clasificar automáticamente determinadas palabras de un texto en categorías predefinidas como personas, organizaciones, lugares, marcas, cantidades, entre otras.
- **Análisis de opinión:** Utiliza técnicas de PLN para averiguar la opinión de un texto subjetivo, clasificándolo en positivo, negativo o neutro.
- **Generación de texto:** Se centra en la aplicación de técnicas de PLN para generar texto coherente y significativo utilizando diversos contextos y estilos.

Estas son solo algunas de las tareas del PLN, pero existen multitud de aplicaciones. Es importante tener en cuenta que el PLN no se limita únicamente al texto escrito. También aborda desafíos complejos en reconocimiento del habla para generar la transcripción de una muestra de audio, o síntesis de voz para comprender un mensaje de audio.

El PLN aprovecha el poder del aprendizaje automático para analizar y entender el contenido generado por humanos mediante un enfoque estadístico. Durante su entrenamiento, los modelos de PLN reciben ejemplos de palabras y frases contextualizadas, junto con sus respectivas interpretaciones. Por ejemplo, al principio, un modelo de PLN puede tener dificultades para distinguir si la palabra "naranja" se refiere al color o a la fruta. Sin embargo, tras ser expuesto a una gran cantidad de ejemplos, como "Me comí una naranja" o "Este coche es de color naranja", el modelo puede aprender a diferenciar los distintos significados de la palabra.

En nuestro caso, nos centraremos en la generación de lenguaje mediante la implementación de un agente conversacional o chatbot basado en los grandes modelos del lenguaje (LLM - Large Language Models). Así pues, nuestro chatbot tendrá la capacidad de aprender y desarrollar frases en respuesta a la información que recibe por parte del usuario, lo que resulta en una experiencia conversacional más natural y fluida considerándose así más cercano y amigable para el usuario. En cualquier caso, queremos destacar que este sistema no se debe utilizar como un sustituto a psiquiatras expertos en la materia.

2.3. Aprendizaje Profundo y Redes Neuronales

El aprendizaje profundo es un campo de la IA que enseña a las computadoras a procesar datos de una manera inspirada en el cerebro humano. Los modelos de aprendizaje profundo pueden reconocer patrones en datos complejos, como imágenes, texto, y sonido, para generar información y predicciones precisas. Las redes neuronales son la tecnología subyacente en el aprendizaje profundo.

2.3.1. Funcionamiento de las redes neuronales

Una red neuronal es un modelo computacional inspirado en la estructura y el funcionamiento del cerebro humano. Se trata de un proceso de aprendizaje profundo, en inglés conocido como *deep learning*, que usa nodos o neuronas interconectados en una estructura de capas que imita al cerebro humano [14]. Así pues, una red neuronal básica está formada por neuronas interconectadas en tres capas:

- **Capa de entrada:** se encarga de procesar los datos con los que la red se alimenta.
- **Capa oculta:** las redes neuronales pueden tener una gran cantidad de estas capas. Cada una de ellas analiza la salida de la capa anterior, la procesa y la pasa a la siguiente capa.
- **Capa de salida:** proporciona el resultado final de todo el procesamiento de datos que realiza la red.

En la Figura 2.1, podemos observar una red neuronal artificial con las distintas capas, donde cada círculo representa las neuronas.

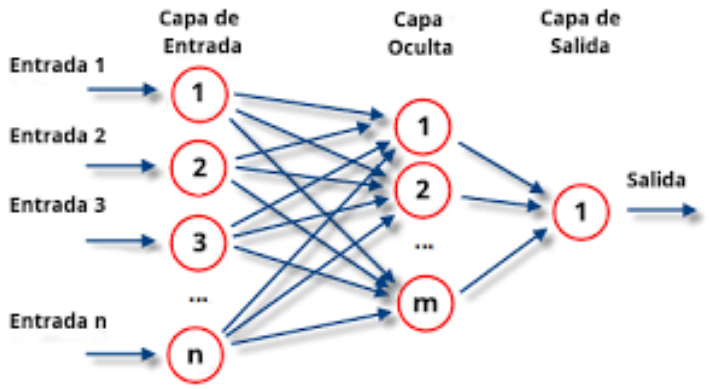

Figura 2.1: Red neuronal artificial

De esta manera, cada neurona tiene una función de activación con un peso asociado a cada conexión, y un sesgo adicional que es compartido entre todas las neuronas de una misma capa. El resultado de la combinación lineal de las entradas ponderadas se envía a la función de activación, y el valor obtenido se transmite a la siguiente capa. En la Figura 2.2, podemos observar el funcionamiento de esta.

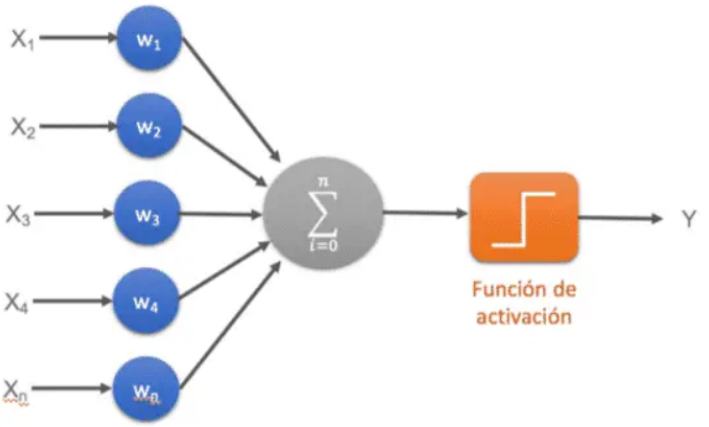

Figura 2.2: Funcionamiento de una neurona

2.4. Tecnologías utilizadas

En este proyecto hemos hecho uso de diferentes tecnologías para desarrollar nuestro chatbot BoTCA:

- **Python [15]:** Es un lenguaje de programación muy utilizado hoy en día, sobre todo en las aplicaciones web, el desarrollo de software, la ciencia de datos, el aprendizaje automático y el PLN.

Figura 2.3: Logo de python

Es usado por los desarrolladores porque es eficiente y fácil de aprender, además de que se puede ejecutar en muchas plataformas distintas. Python cuenta con una gran biblioteca estándar que contiene códigos reutilizables para casi cualquier tarea. Por ello, hemos optado por este lenguaje. Entre las bibliotecas que ofrece, en este proyecto hemos hecho uso de las siguientes:

- `Transformer [16]`: Proporciona una Interfaz de Programación de Aplicaciones (API - Application Programming Interface) facilitando la descarga y el entrenamiento de una gran variedad de modelos. Se puede realizar un pre-entrenamiento del modelo elegido para adaptarlo a tareas de distintas modalidades, en este caso, para la generación de respuestas por parte del sistema.
- `FastAPI [17]`: FastAPI es un moderno marco de desarrollo web para construir APIs (interfaces de programación de aplicaciones) en Python de manera rápida, fácil y eficiente. Está diseñado para ser de alto rendimiento, fácil de usar y con una sintaxis declarativa que permite a los desarrolladores crear rápidamente servicios web. Por esto, hemos considerado oportuno su uso para el desarrollo de BoTCA.
- `Uvicorn [18]`: Una implementación de servidor web *Asynchronous Server Gateway Interface* (ASGI) para Python. Hasta hace poco, Python carecía de una

interfaz mínima de servidor/aplicación de bajo nivel para los marcos asíncronos. ASGI es una especificación que define cómo las aplicaciones web asincrónicas interactúan con los servidores web y otros componentes del ecosistema web. Permite la creación de aplicaciones web asíncronas eficientes y escalables al proporcionar una interfaz común para manejar solicitudes y respuestas de forma asíncrona.

- `LangChain` [19]: Es un marco de trabajo para desarrollar aplicaciones impulsadas por modelos de lenguaje. Permite aplicaciones que tienen en cuenta el contexto, proporcionando herramientas y abstracciones para mejorar la personalización, precisión y relevancia de la información que generan los modelos.
- `PyPDF` [20]: Es una biblioteca de Python para trabajar con archivos PDF, permitiendo leer, extraer texto y realizar operaciones de manipulación como dividir o fusionar archivos PDF.
- `Docx2TXT` [21]: Es una herramienta que extrae texto con formato de archivos .docx de Microsoft Word, convirtiéndolo en texto plano. Es muy útil para acceder al contenido de documentos Word sin necesidad de tener instalado Microsoft Word.
- `Python-multipart` [22]: Es una biblioteca para manejar solicitudes POST multipart/form-data. Esto es especialmente útil al ejecutar un servidor web donde los usuarios pueden cargar archivos a través de formularios en línea. Permite procesar y manipular los datos enviados con las solicitudes HTTP que contienen archivos adjuntos.
- `OpenAI` [23]: Es una herramienta que facilita el acceso a la API de OpenAI desde aplicaciones escritas en Python. Permite interactuar con los servicios de OpenAI para realizar tareas como generación de texto, clasificación de texto, preguntas y respuestas, entre otras. Proporciona definiciones de tipos para los parámetros de solicitud y los campos de respuesta, lo que simplifica el proceso de comunicación con la API. Además, ofrece clientes tanto síncronos como asíncronos para adaptarse a diferentes necesidades de desarrollo. Es útil para integrar la tecnología de OpenAI en proyectos de PLN y otras aplicaciones que requieran inteligencia artificial basada en texto.
- `ChromaDB` [24]: Chroma es una base de datos de embedding de código abierto. Facilita la construcción de aplicaciones de LLM al permitir que el conocimiento, los hechos y las habilidades sean fácilmente adaptables para los modelos de lenguaje. Proporciona las herramientas que permite almacenar incrustaciones y su metadatos, incrustar documentos y consultas, y buscar embeddings ya almacenados.
- `Tiktoken` [25]: Es un tokenizador Byte Pair Encoding (BPE) diseñado para su uso con los modelos de OpenAI. El BPE permite convertir texto en tokens.

Una vez presentadas las distintas bibliotecas que hemos necesitado para la realización de este trabajo, vamos a pasar a explicar qué servicios web y plataformas de computación hemos usado:

- **Visual Studio Code [26]:** Es un entorno de desarrollo integrado (IDE - *Integrated Development Enviroment*) creado por Microsoft para desarrollar aplicaciones de software en diversos lenguajes como C, C++, Visual Basic, Python, entre otros. Ofrece herramientas para escribir, depurar, compilar y mantener código de manera eficiente. Incluye características como resaltado de sintaxis, autocompletado de código, depuración visual, administración de proyectos y control de versiones.

Figura 2.4: Logo de Visual Studio Code

- **Google Colaboratory [27]:** Es una herramienta que nos permite programar y ejecutar Python desde el navegador, sin necesidad de configurar nada y con acceso gratuito a GPUs. Esto es muy útil para hacer un entrenamiento de modelos más rápido que mediante el uso de CPU, aunqe el acceso a la GPU tiene algunas limitaciones de espacio y tiempo.

Figura 2.5: Logo de Google Colaboratory

De todas estas herramientas y tecnologías, hay una que destacamos para el desarrollo de nuestro proyecto, los **Transformers**. Por ello, vamos a dedicar una sección adicional para profundizar en su explicación.

2.5. Transformers

Los modelos de Transformer han revolucionado la forma en que abordamos las tareas de PLN y han mejorado significativamente la capacidad de las máquinas para comprender y generar texto de manera más precisa y contextualmente relevante. Los `Transformers` [28] son una estructura de redes neuronales que ha revolucionado el campo del PLN y ha mejorado considerablemente la capacidad de los LLM. Estos permiten a los modelos aprender conexiones complejas entre palabras y oraciones, lo que resulta en una comprensión y generación de texto en lenguaje natural más precisa y avanzada. Este término se definió por primera vez en 2017 en el artículo 'Attention is all you need' [29]. El trabajo original se centraba en tareas de traducción, aunque hoy en día se usen en la mayoría de tareas de PLN. Tras la aparición de los Transformers, aparecen varios modelos que han tenido un gran impacto en el área (Figura 2.6):

- **GPT (Junio de 2018):** el primer modelo de Transformers preentrenados, fue GPT (Generative Pretrained Transformer), que se utilizó para ajustar varias tareas de PLN y logró resultados líderes en su campo.
- **BERT (Octubre de 2018):** BERT (Bidirectional Encoder Representations from Transformers) es una técnica basada en redes neuronales para el pre-entrenamiento del PLN desarrollada por Google. Lo que hace único a BERT es su capacidad para captar el contexto bidireccional en una oración, lo que significa que puede entender el significado de una palabra en relación con las palabras que la rodean.
- **GPT-2 (Febrero de 2019):** una versión mejorada y más grande de GPT, que inicialmente no se lanzó al público debido a preocupaciones éticas.
- **DistilBERT (Octubre de 2019):** una versión más ligera y rápida de BERT que conserva el 97 % del rendimiento de BERT.
- **BART y T5 (Octubre de 2019):** dos modelos preentrenados de gran escala que utilizan la misma arquitectura del modelo original de Transformer.
- **GPT-3 (Mayo de 2020):** una versión aún más grande de GPT-2 que demostró un buen rendimiento en una amplia gama de tareas sin necesidad de ajustes adicionales (conocido como aprendizaje zero-shot).

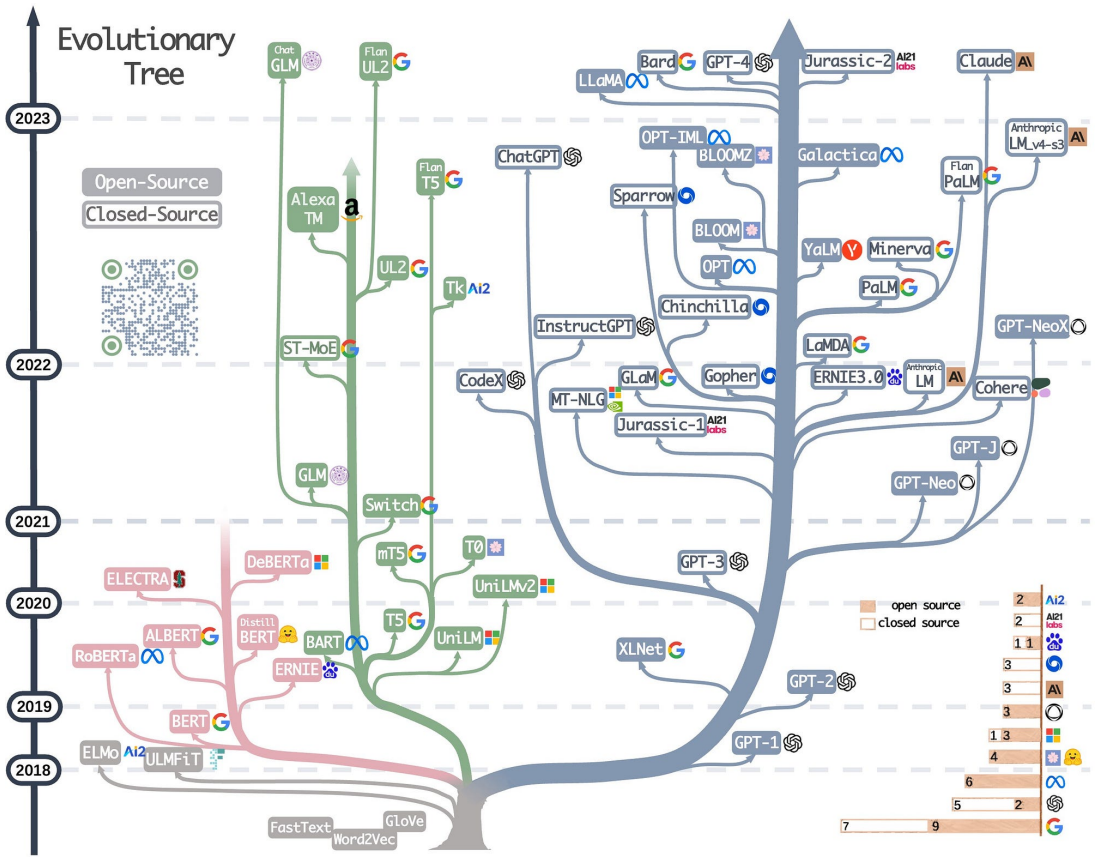

Figura 2.6: Árbol de evolución de Transformers

Esta lista pretende destacar algunos de los diferentes modelos de Transformers. En general, estos modelos se pueden agrupar en tres categorías:

- Modelos similares a GPT (también conocidos como modelos auto-regresivos).
- Modelos similares a BERT (también conocidos como modelos auto-codificadores).
- Modelos similares a BART/T5 (también conocidos como modelos de secuencia a secuencia).

Todos los modelos de Transformers mencionados anteriormente han sido entrenados como modelos de lenguaje. Esto implica que han sido entrenados utilizando grandes volúmenes de texto sin procesar de manera auto-supervisada. El aprendizaje auto-supervisado es un tipo de entrenamiento en el que el objetivo se deriva automáticamente de las entradas del modelo. Esto significa que no se requiere la intervención humana para etiquetar los datos.

Estos modelos desarrollan un entendimiento estadístico del lenguaje en el que fueron entrenados, pero no son particularmente útiles para tareas prácticas específicas. Por lo

tanto, el modelo general preentrenado se somete a un proceso llamado transferencia de aprendizaje (o 'transfer learning' en inglés). Durante este proceso, el modelo se ajusta de manera supervisada, es decir, utilizando etiquetas proporcionadas por humanos, para una tarea específica.

Un ejemplo de una tarea es predecir la siguiente palabra en una oración basándose en las palabras anteriores. Esto se conoce como modelado de lenguaje causal, ya que la salida depende de las entradas pasadas y presentes, pero no de las futuras. Podemos ver un ejemplo en la Figura 2.7

Otro ejemplo es el modelado de lenguaje oculto, en el que el modelo predice una palabra oculta en la oración. La Figura 2.8 muestra un ejemplo de este tipo.

Salvo en algunos casos atípicos, como DistilBERT, la estrategia general para mejorar el rendimiento consiste en aumentar el tamaño de los modelos, así como la cantidad de datos utilizados para su preentrenamiento.

Desafortunadamente, el entrenamiento de un modelo, especialmente uno de gran tamaño, demanda enormes cantidades de datos. Esto se convierte en un proceso costoso en términos de tiempo y recursos de computación, lo que puede tener incluso un impacto ambiental considerable, como se ilustra en la Figura 2.9

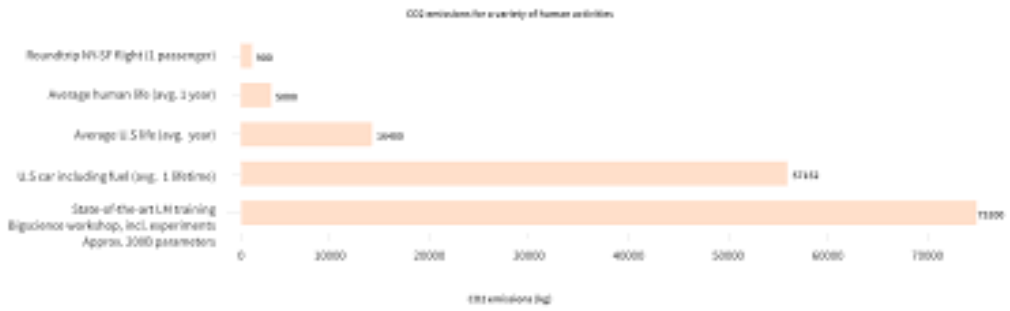

Figura 2.9: Ejemplo de modelado de lenguaje oculto

Por otra parte, los autores de BERT introdujeron un segundo aporte: `la transferencia de aprendizaje` [30]. El preentrenamiento implica entrenar un modelo desde cero, donde los pesos se inicializan aleatoriamente y el proceso de entrenamiento comienza sin ningún conocimiento previo.

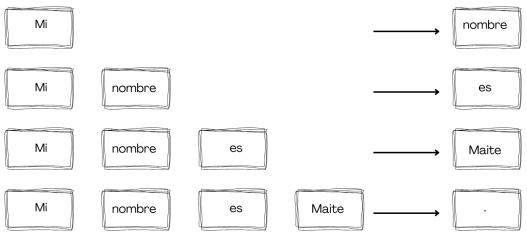

Figura 2.7: Ejemplo de modelado de lenguaje casual

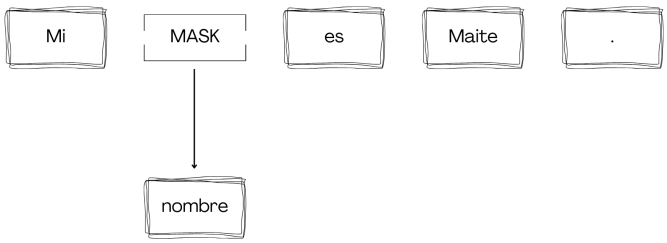

Figura 2.8: Ejemplo de modelado de lenguaje oculto

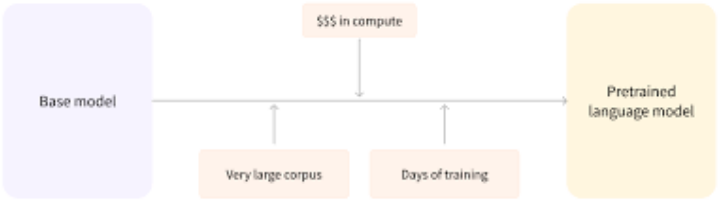

Figura 2.10: Proceso de preentrenamiento

Normalmente se realiza sobre conjuntos de datos extensos, lo que requiere un gran corpus de datos y puede llevar varias semanas de entrenamiento.

Por otro lado, el ajuste (o fine-tuning) es el proceso de entrenamiento que ocurre después de que el modelo ha sido preentrenado. Durante el ajuste, comenzamos con un modelo de lenguaje preentrenado y realizamos un aprendizaje adicional utilizando un conjunto de datos específico para la tarea en cuestión. Entonces, la cuestión es por qué no entrenar directamente

para la tarea final. Hay varias razones:

1. El modelo preentrenado ya ha sido entrenado con un conjunto de datos similar al del ajuste. Por lo tanto, el proceso de ajuste puede aprovechar el conocimiento adquirido por el modelo inicial durante el preentrenamiento. Por ejemplo, para problemas de procesamiento del lenguaje natural, el modelo preentrenado ya tendrá cierto entendimiento estadístico del idioma que se está utilizando para la tarea.

2. Dado que el modelo preentrenado ha sido entrenado con una gran cantidad de datos, el ajuste requiere menos datos para obtener resultados decentes.

3. Por la misma razón, el tiempo y los recursos necesarios para obtener buenos resultados son mucho menores.

Así, si aprovechamos un modelo preentrenado y lo ajustamos con un corpus de datos, el ajuste solo requeriría una cantidad limitada de datos, ya que el conocimiento adquirido por el modelo preentrenado se "transfiere", de ahí el término transferencia de aprendizaje.

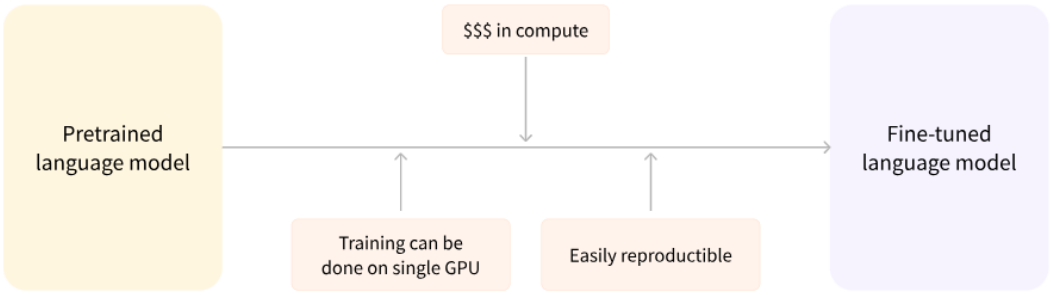

Figura 2.11: Ajuste más económico en tiempo, datos, recursos financieros y ambientales

De este modo, el ajuste de un modelo es más económico en términos de tiempo, datos, recursos financieros y ambientales. Además, es más rápido y fácil de iterar en diferentes esquemas de ajuste, ya que el proceso de entrenamiento es menos restrictivo que el preentrenamiento completo. Este enfoque también tiende a producir mejores resultados que entrenar un modelo desde cero, a menos que se disponga de una gran cantidad de datos. Por lo tanto, es más eficiente aprovechar un modelo preentrenado lo más cercano posible a la tarea específica y ajustarlo en consecuencia.

Arquitectura general de los Transformers

Un modelo está formado por dos bloques:

- **Codificador:** este recibe una entrada y construye una representación de sus características, lo que significa que el modelo está optimizado para lograr un entendimiento dada una entrada.
- **Decodificador:** en este caso, se usa la representación del codificador junto con otras entradas para generar una secuencia objetivo. Esto implica que el modelo está preparado para generar salidas.

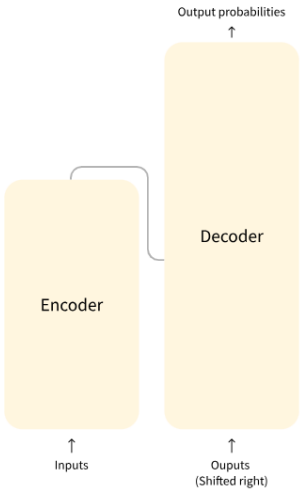

Figura 2.12: Bloques de la arquitectura original

Cada uno de estos bloques podemos usarlo independientemente, dependiendo de la tarea que queramos realizar:

- **Modelos con solo codificadores:** Buenos para las tareas que requieren el entendimiento de la entrada, como la clasificación de oraciones y reconocimiento de entidades nombradas.
- **Modelos con solo decodificadores:** Buenos para tareas generativas como la generación de textos.
- **Modelos con codificadores y decodificadores o modelos secuencia a secuencia:** Buenos para tareas generativas que requieren una entrada, como la traducción o resumen.

Capas de atención

Los Transformers incluyen capas especiales conocidas como capas de atención, destacadas en el trabajo titulado 'Attention Is All You Need' [29]. Estas capas son fundamentales para

el modelo, ya que indican dónde debe enfocarse al procesar cada palabra de una oración, ignorando partes menos relevantes.

Por ejemplo, al traducir del inglés al francés, el modelo debe considerar palabras cercanas para entender correctamente el contexto y conjugar verbos de manera adecuada. El significado de una palabra está fuertemente influenciado por el contexto, ya sea palabras anteriores o posteriores en la oración. Este principio se aplica a cualquier tarea de procesamiento de lenguaje natural: comprender una palabra implica considerar su contexto para captar su significado completo.

Arquitectura original

La estructura del Transformer fue inicialmente concebida para el propósito de la traducción. Durante el proceso de entrenamiento, el codificador recibe las entradas (oraciones) en un idioma específico, mientras que el decodificador recibe esas mismas oraciones pero en el idioma de destino.

En el codificador, las capas de atención pueden considerar todas las palabras en una oración, ya que la traducción de una palabra puede depender del contexto de la oración completa. Por otro lado, el decodificador trabaja de manera secuencial y solo puede atender a las palabras previamente traducidas en la oración (es decir, las palabras antes de la que está siendo generada). Por ejemplo, cuando se han predicho las primeras tres palabras del objetivo de traducción, estas se proporcionan al decodificador, que luego utiliza toda la información del codificador para intentar predecir la siguiente palabra.

Para agilizar el proceso de entrenamiento (cuando el modelo tiene acceso a las oraciones objetivo), se le suministra al decodificador el objetivo completo, pero sin información sobre palabras futuras. Por ejemplo, al intentar predecir la cuarta palabra, la capa de atención del decodificador solo puede considerar las palabras en las posiciones 1 a 3.

Como podemos ver en la figura 2.13, la disposición original del Transformer se presentaba con el codificador a la izquierda y el decodificador a la derecha.

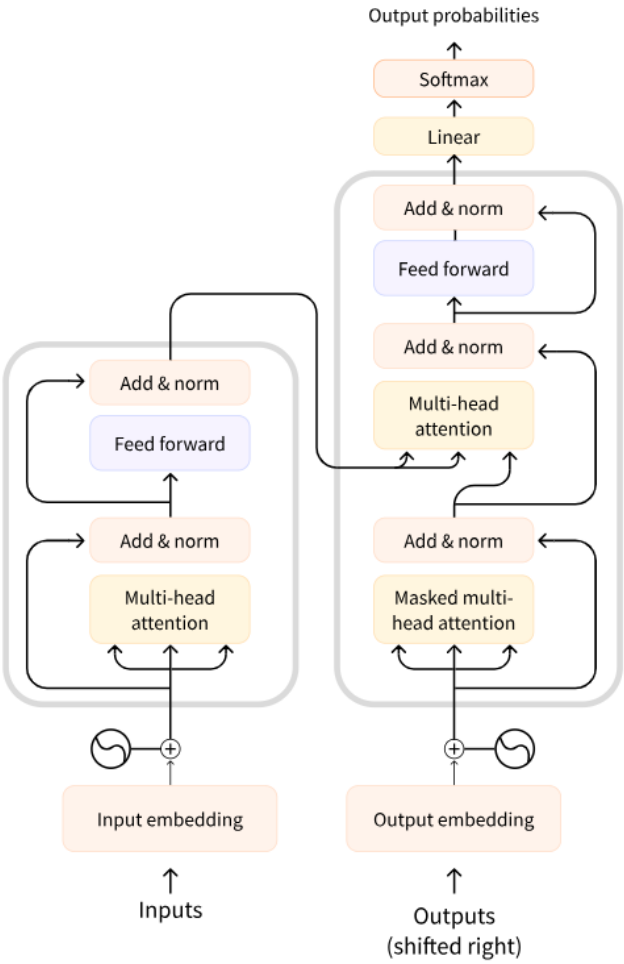

Figura 2.13: Disposición original del Transformer

Podemos observar que la primera capa de atención en un bloque de decodificador se enfoca en todas las entradas anteriores al decodificador, mientras que la segunda capa de enfoque utiliza la salida del codificador.

Esto le permite acceder a toda la oración de entrada para predecir de manera más precisa la palabra actual. Siendo esto muy útil ya que diferentes idiomas pueden tener reglas gramaticales que ordenan las palabras de manera diferente, o el contexto que se proporciona después puede ser útil para determinar la mejor traducción de una palabra dada.

La máscara de atención también se puede utilizar en el codificador/decodificador para evitar que el modelo preste atención a algunas palabras especiales, como la palabra de relleno que se utiliza para que todas las entradas tengan la misma longitud cuando se agrupan oraciones.

2.6. Modelos del lenguaje

El juego de Shannon es útil para explicar el concepto de modelo de lenguaje. En lugar de predecir el siguiente carácter en un texto, imaginemos que queremos predecir la siguiente palabra. En lugar de consultar a una persona, usamos un ordenador al que le proporcionamos las palabras previamente adivinadas, llamadas çontexto". Mediante técnicas matemáticas, un modelo de lenguaje puede determinar cuál es la palabra más probable de aparecer dada cierta secuencia de palabras, es decir, el contexto.

En términos más formales, dado un contexto específico, un modelo de lenguaje puede asignar una probabilidad a cada palabra posible que pueda aparecer a continuación. Este usa el aprendizaje automático, anteriormente explicado, para hacer una distribución de probabilidad sobre las palabras utilizadas para predecir la siguiente palabra más probable en una oración en función de la entrada anterior.

Los `Modelos de Lenguaje de Gran Escala` [31], Large Language Model en inglés (LLM), son sistemas de inteligencia artificial, que se engloban dentro del ámbito del PLN, diseñados para comprender y generar texto en lenguaje natural. Un ejemplo popular que usa estos modelos es ChatGPT, del cual hablaremos posteriormente. Estos sistemas tienen la capacidad de interpretar y generar texto en una variedad de idiomas y contextos, lo que los hace herramientas muy poderosas en áreas como la traducción automática, la generación de resúmenes y la creación de contenido.

En este proyecto, hemos hecho uso de tres modelos que están a la orden del día, y a los que hemos dedicado tres subapartados, pero antes, vamos a explicar alguno de los más usados actualmente [32]:

- **GPT 3.5 [33]:** El Transformer Pre-entrenado Generativo (GPT) 3.5, desarrollado por OpenAI, es un modelo del lenguaje que ha llevado el PLN a nuevos niveles. Sus redes neuronales son capaces de entender y generar texto similar al humano, lo que las hace excepcionalmente versátiles en diversas aplicaciones. Puede generar frases, párrafos e incluso artículos completos con un estilo que refleja la inteligencia humana.

Figura 2.14: GPT 3.5

- **GPT 4 [33]:** Esta última iteración de OpenAI cuenta con mejoras drásticas sobre las capacidades de PLN de GPT 3.5. Se considera el modelo más grande del mercado. De los dos modelos GPT, este no solo comprende y genera mejor texto, sino que tiene la capacidad de procesar imágenes y vídeos, lo que la hace más versátil.

Figura 2.15: GPT 4

- **Gemini [34]:** Este es un nuevo chatbot de LLM desarrollado por Google AI. Está entrenado en un enorme conjunto de datos de texto y código. Esto lo hace capaz de producir texto, traducir varios idiomas, crear código, generar contenido variado y proporcionar respuestas informativas a las preguntas. Además, también puede acceder a datos del mundo real a través de Google Search, lo que le permite comprender y abordar un espectro más amplio de prompts y consultas.

Figura 2.16: Gemini

- **LLaMA [35]:** Es un LLM de código abierto desarrollado por Meta AI que aún está en desarrollo. Está diseñado para ser un LLM versátil y potente que se puede utilizar

para diversas tareas, incluyendo la resolución de consultas, la comprensión del lenguaje natural y la comprensión de lectura.

Figura 2.17: LLaMA

- **Cohere [36]:** Es un modelo desarrollado por una startup canadiense con el mismo nombre. Este LLM de código abierto está entrenada en base a un conjunto de datos diverso e inclusivo, lo que lo convierte en un experto en manejar numerosos idiomas y acentos. Además, los modelos de Cohere están entrenados en base a un corpus de texto grande y diverso, lo que los hace más efectivos para manejar una amplia gama de tareas.

Figura 2.18: Cohere

- **Gemma[37]:** Desarrollado por Google DeepMind y otros equipos de IA de Google, Gemma aprovecha las técnicas aprendidas durante el desarrollo de Gemini. A diferencia de este, los modelos de Gemma se ejecutan localmente en una computadora de escritorio. Es el primer lanzamiento significativo de Google de un LLM abierto desde que ChatGPT inició el frenesí de los chatbots de IA en 2022.

Figura 2.19: Gemma

Estos son solo algunos de los principales modelos en 2024, aunque existe una gran variedad de LLMs usados para distintas tareas.

2.6.1. GPT

Ahora podemos discutir la tecnología que respalda ChatGPT. `GPT` es una abreviatura de preentrenamiento generalizado (*Generalized Pre-Training*, en inglés).

Figura 2.20: logo de OpenAI

El modelo GPT [38], se basa en la parte decodificadora de la arquitectura Transformer. Es un modelo de lenguaje que produce la siguiente palabra a partir de un contexto. Una característica clave de GPT es que es un modelo 'preentrenado', lo que implica que los parámetros del modelo han sido previamente ajustados con una gran cantidad de datos antes de que un usuario comience a trabajar con él. A partir de este punto, el usuario puede usar simplemente el modelo para predecir la siguiente palabra, o modificarlo para realizar diferentes tareas de procesamiento de lenguaje natural. La base de este enfoque es que el amplio conocimiento integrado en el modelo preentrenado se puede transferir y aplicar a otra tarea relacionada. Este proceso se conoce como aprendizaje por transferencia, como ya mencionamos en anteriores apartados.

GPT representa un ejemplo de un nuevo enfoque en el campo de la IA: los modelos fundacionales [39]. Un modelo fundacional es un modelo entrenado a partir de grandes conjuntos de datos que puede adaptarse mediante técnicas de aprendizaje por transferencia para hacer una gran cantidad de tareas. Estos son por naturaleza incompletos, debiéndose adaptar a la tarea para la que se van a usar.

La aparición de nuevas arquitecturas de aprendizaje profundo como GPT marcó el comienzo reciente de los modelos fundacionales. Estos modelos están vinculados a dos conceptos: homogeneización y emergencia [39].

- El concepto de **homogeneización** se refiere al hecho de que un solo tipo de modelo puede ser utilizado para resolver una amplia variedad de tareas. Desde la llegada de la arquitectura Transformer, la homogeneización en el campo de la inteligencia artificial ha experimentado un avance significativo. Los Transformers no solo se aplican con éxito en tareas de PLN, sino también en una amplia gama de tareas que implican análisis de imágenes o secuencias biológicas.

- El segundo concepto es la **emergencia**, que implica que el comportamiento de estos modelos surge del proceso de aprendizaje, en lugar de estar predefinido. Esto significa que un modelo fundacional puede mostrar capacidades que no se habían previsto al momento de su creación. Si bien esto puede ser fascinante desde el punto de vista científico, también puede generar preocupación si estas capacidades no previstas pueden ser perjudiciales o utilizadas de manera maliciosa.

2.6.2. Llama

Otro de los grandes modelos del lenguaje, es el sistema de inteligencia artificial creado por Meta.

Figura 2.21: Logo de META

`LLaMA` son las singlas de `Large Language Model Meta AI`. Como indica su nombre, es un modelo de lenguaje por IA, así es un competidor de GPT, anteriormente explicado. Para Meta, es un pequeño modelo que asegura requerir menos potencia y recursos informáticos para conseguir lo que llaman resultados competitivos". Entonces, podemos decir que posiblemente obtenga resultados menos completos pero será un modelo más eficiente, lo que ayudará a que sea utilizado por entidades con menos recursos [40].

Para la primera versión, se entrenaron cuatro tamaños de modelos: 7, 13, 33 y 65 mil millones de parámetros. Según los desarrolladores de LLaMA, el rendimiento del modelo de parámetros 13B sobrepasó al del GPT-3 en la mayoría de los puntos de referencia de PNL, a pesar de que este último es significativamente más grande, con 175 mil millones de parámetros. Además, se encontró que el modelo más grande era competitivo con otros modelos de última generación como PaLM y Chinchilla [41].

Tradicionalmente, los LLM más potentes han estado disponibles solo a través de API restringidas, pero Meta optó por liberar los pesos del modelo de LLaMA para la comunidad de investigación bajo una licencia no comercial [42].

En julio de 2023, introdujo varios modelos nuevos, entre ellos Llama 2, que utilizan diferentes cantidades de parámetros, incluyendo 7, 13 y 70 mil millones.

Figura 2.22: LLaMA-2

Así, `LLaMA-2` fue anunciado por Meta en asociación con Microsoft, siendo la siguiente generación de LLaMA. Esta usa la arquitectura Transformer, la cual es estándar para el modelado de lenguajes desde 2018. La arquitectura del modelo se mantuvo casi idéntica a la de los modelos LLaMA-1, sin embargo, se incrementó el conjunto de datos de entrenamiento en un 40 % para los modelos fundamentales.

2.6.3. Gemma

Este último apartado lo dedicaremos al nuevo modelo lanzado por Google.

Figura 2.23: Logo de Gemma, el nuevo modelo lanzado por Google

`Gemma` [43] es una familia de modelos abiertos, ligeros y de última generación, construidos a partir de la misma investigación y tecnología utilizadas para crear los modelos Gemini. Desarrollados por Google DeepMind y otros equipos de Google, Gemma está inspirado en Gemini.

Algunos aspectos importantes de este nuevo modelo son:

- Existencia de pesos de modelos en dos tamaños: Gemma 2B y Gemma 7B. Cada tamaño se lanza con variantes pre-entrenadas y ajustadas según instrucciones.
- Herramientas para inferencia y ajuste fino supervisado (SFT) en todos los principales frameworks: JAX, PyTorch y TensorFlow a través de Keras 3.0 nativo.
- Notebooks de Colab y Kaggle listos para usar, junto con la integración con herramientas populares como Hugging Face, MaxText, NVIDIA NeMo y TensorRT-LLM, facilitan el inicio con Gemma.

Los modelos Gemma comparten componentes técnicos y de infraestructura con Gemini. Esto permite que Gemma 2B y 7B alcancen un rendimiento líder en su clase para sus tamaños en comparación con otros modelos abiertos. La siguiente gráfica 2.24 muestra el rendimiento de Gemma en benchmarks comunes, comparado con Llama-2 7B y 13B.

CAPABILITY	BENCHMARK	DESCRIPTION	Gemma 7B	Llama-2 7B	Llama-2 13B
General	MMLU 5-shot, top-1	Representation of questions in 57 subjects (incl. STEM, humanities and others)	64.3	45.3	54.8
Reasoning	BBH -	Diverse set of challenging tasks requiring multi-step reasoning	55.1	32.6	39.4
	HellaSwag 0-shot	Commonsense reasoning for everyday tasks	81.2	77.2	80.7
Math	GSM8K maj@1	Basic arithmetic manipulations (incl. Grade School math problems)	46.4	14.6	28.7
	MATH 4-shot	Challenging math problems (incl. algebra, geometry, pre-calculus, and others)	24.3	2.5	3.9
Code	HumanEval pass@1	Python code generation	32.3	12.8	18.3

Figura 2.24: Gemma vs LlaMA-2

Capítulo 3

Análisis

3.1. Requisitos del sistema

3.1.1. Funcionales

Como hemos explicado anteriormente, los TCA son enfermedades mentales que están presentes en la vida de muchas personas. Por ello, hemos intentado con este chatbot, BoTCA, proporcionar información para aquellas personas que sufren, o conocen a otras que estén pasando por este trastorno. No obstante, hay que destacar que la intención de la creación de este chatbot no es reemplazar la terapia psicológica por expertos, sino propocionar información confiable sobre esta enfermedad mental. Por consiguiente, el principal requisito es **realizar consulta**. Así pues, el actor pregunta sobre los TCA al sistema, y este debe generar una respuesta acorde a la necesidad de información del usuario.

3.1.2. No funcionales

Los requisitos no funcionales se refieren a cómo el sistema debe comportarse y sus características de calidad:

1. **Disponibilidad:** debe estar disponible 24/7 para proporcionar información en cualquier momento que el usuario lo necesite.
2. **Escalabilidad:** debe ser capaz de manejar un gran número de consultas simultáneas sin degradar su rendimiento.
3. **Seguridad:** debe garantizar la privacidad y seguridad de los datos del usuario, cumpliendo con las normativas de protección de datos.
4. **Fiabilidad:** debe ofrecer respuestas precisas y confiables basadas en fuentes verificadas y actualizadas sobre los TCA.

5. **Usabilidad:** debe ser fácil de usar para personas de todas las edades y niveles de competencia tecnológica.

6. **Mantenimiento:** debe permitir actualizaciones y mantenimiento regular para asegurar la inclusión de nueva información y mejoras en la funcionalidad.

7. **Rendimiento:** debe responder a las consultas de los usuarios en un tiempo mínimo para garantizar una experiencia fluida.

3.2. Casos de uso

Una vez que hemos identificado los requisitos y necesidades que BoTCA debe cumplir, podemos pasar a diseñar los casos de uso.

Un caso de uso describe los pasos o actividades necesarias para llevar a cabo un proceso específico. Las entidades que participan en un caso de uso se conocen como actores. En el contexto de la Ingeniería del Software, un caso de uso representa una secuencia de interacciones entre un sistema y sus actores en respuesta a un evento iniciado por un actor principal sobre el sistema [44].

Los diagramas de casos de uso son herramientas que se utilizan para especificar la comunicación y el comportamiento de un sistema a través de su interacción con usuarios u otros sistemas. Las relaciones son conexiones entre los elementos del modelo, como la especialización y la generalización. Estos diagramas ayudan a ilustrar los requisitos del sistema al demostrar cómo responde a eventos dentro de su alcance.

Así pues, los diagramas de casos de uso son ampliamente utilizados para capturar requisitos funcionales, especialmente con el desarrollo del paradigma de programación orientada a objetos, donde tuvieron su origen. Sin embargo, pueden ser igualmente efectivos con otros paradigmas de programación.

En nuestro caso, tenemos un solo caso de uso especialmente sencillo, pues el usuario debe consultar cualquier duda que tenga sobre los TCA, y el sistema debe ser capaz de responder en base a los documentos que procesa sobre estas enfermedades (Figura 3.1. Además, en la Figura 3.2 se observar una representación de nuestro requisito funcional mediante casos de uso, y más concretamente elaborando las **historias de caso de uso**.

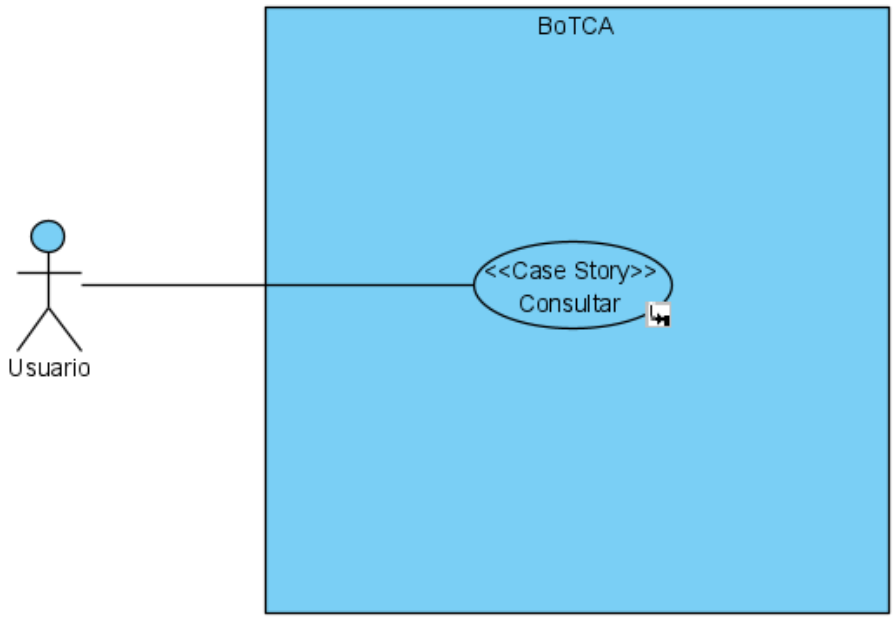

Figura 3.1: Diagrama de historias de caso

	Actor	Caso de uso	Valor de negocio
ID	*Como*	*Quiero*	*Para*
1	Usuario	Consultar	Informarme sobre los TCA

Figura 3.2: Análisis textual de BoTCA

Una vez que tenemos nuestro caso de uso, creamos la descripción textual de este. En la Figura 3.3, podemos ver una tabla donde hemos incluido esta descripción textual asociada a nuestro caso de uso.

Caso de uso	**Consultar**
Actor principal	Usuario
Sistema	BoTCA
Nivel	Objetivo del usuario
Precondición	El usuario no necesita crearse una cuenta/acceso
Flujo Básico	
1	Preguntar
2	Generar respuesta
3	Evaluar respuesta
4	Salir
Flujo alternativo	
2.A	Descargar fichero de preguntas y respuestas evaluadas

Figura 3.3: Historia de nuestro caso de uso

Finalmente, a partir de los análisis textuales e historias de caso, hemos creado nuestro **diagrama de caso** (Figura 3.4).

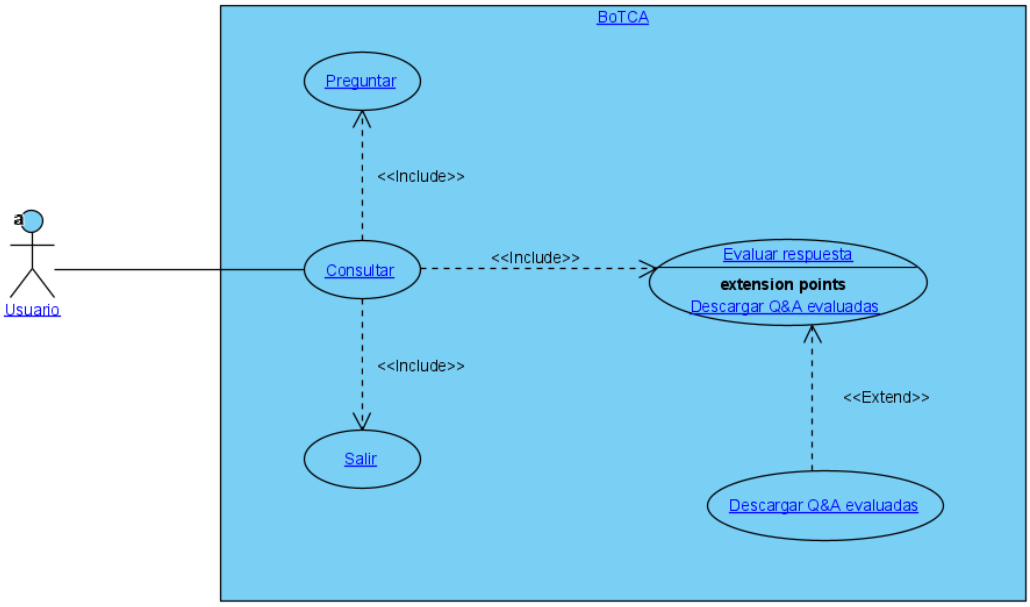

Figura 3.4: Caso de uso de BoTCA

3.3. Ciclo de vida de los objetos de nuestro sistema

El objetivo de este es representar el ciclo de vida de los objetos de nuestro sistema y la posible interacción entre ellos. Así pues, hemos querido representar con un **diagrama de actividades** la secuencia que el actor principal, en nuestro caso un usuario que quiera informarse sobre los TCA, debe realizar (Figura 3.5).

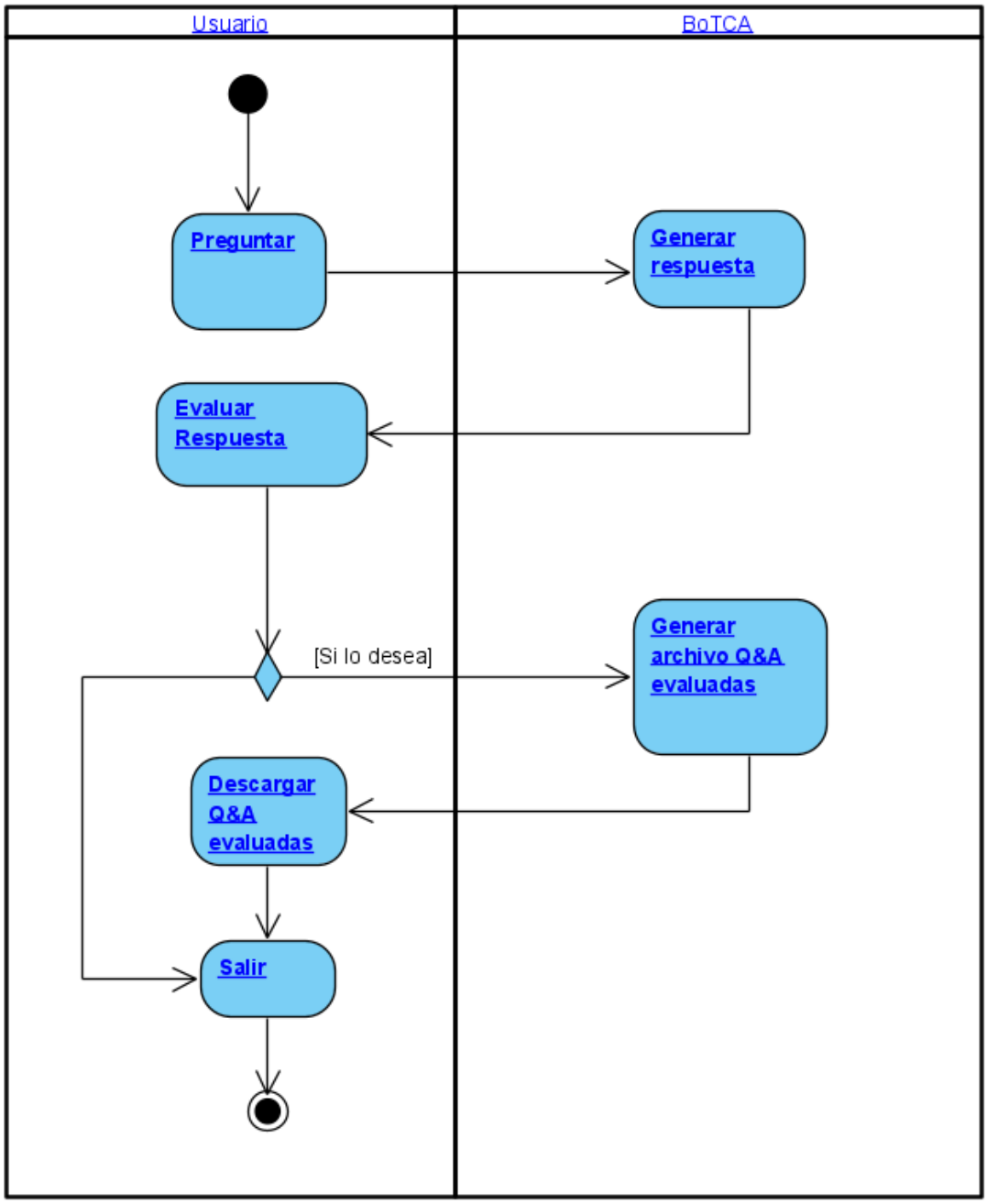

Figura 3.5: Diagrama de actividades de BoTCA

3.4. Diagrama de secuencia

Tras la realización de los diagramas anteriores, hemos elaborado un **diagrama de secuencia**. Este es un tipo de diagrama de interacción porque describe cómo, y en qué orden, un grupo de objetos funcionan en conjunto. En la Figura 3.6 podemos ver nuestro diagrama.

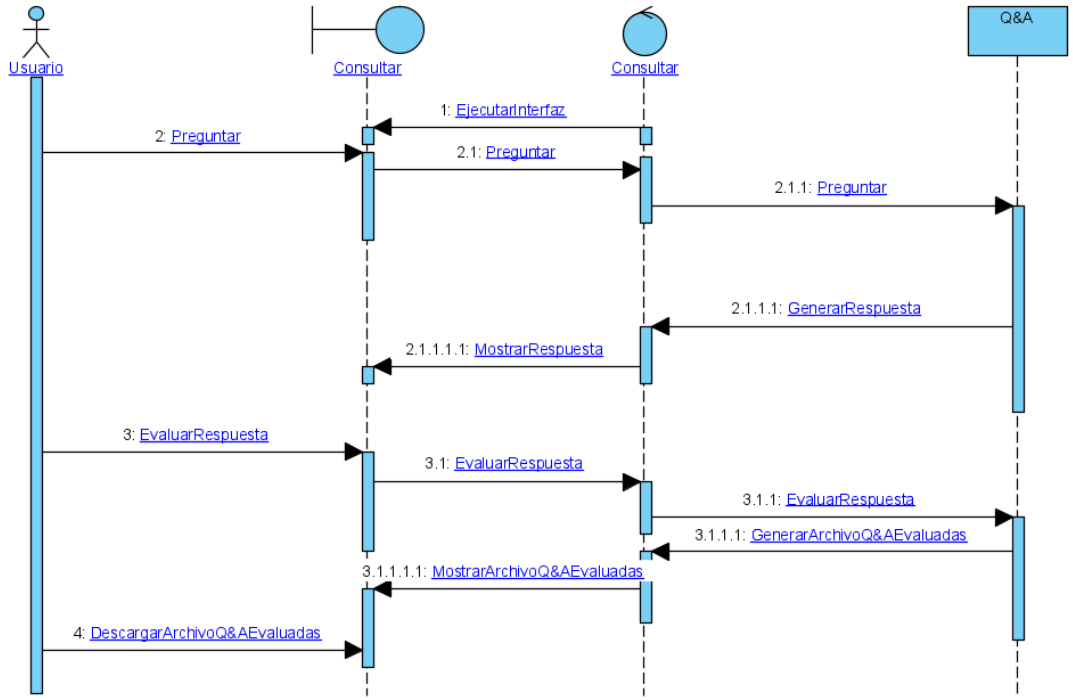

Figura 3.6: Diagrama de secuencia de BoTCA

3.5. Realización de casos de uso

A partir del diagrama anterior, podemos conseguir el **diagrama de comunicación** (Figura 3.7). Un diagrama de comunicación modela las interacciones entre objetos o partes en términos de mensajes en secuencia. Los diagramas de comunicación representan una combinación de información tomada desde el diagrama de clases, secuencia, y diagrama de casos de uso describiendo tanto la estructura estática como el comportamiento dinámico de un sistema [45].

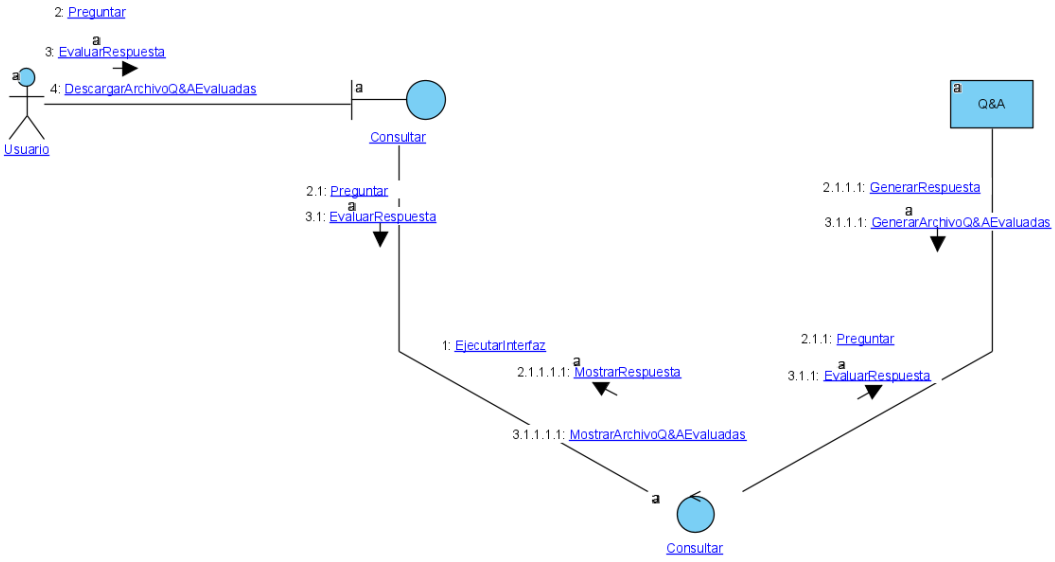

Figura 3.7: Diagrama de comunicación de BoTCA

Además, hemos querido también representar con un **diagrama de clases**, qué objetos de nuestro sistema están relacionados entre sí (Figura 3.8).

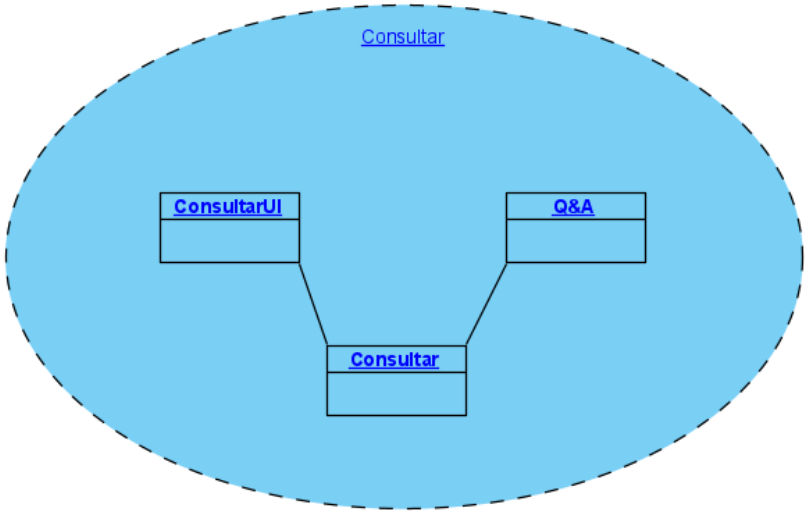

Figura 3.8: Diagrama de colaboración de BoTCA

Por último, hemos generado un único **diagrama de clases de análisis** que representa el funcionamiento general de BoTCA (Figura 3.9).

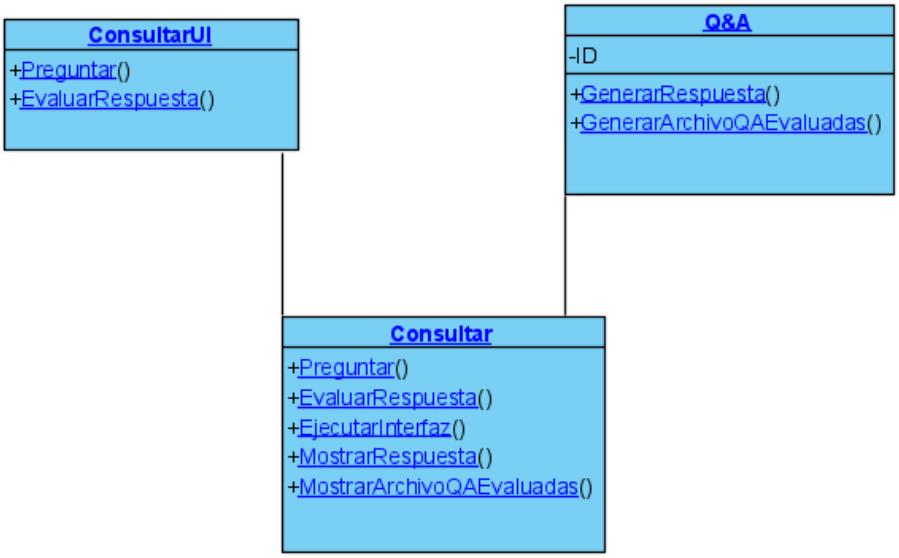

Figura 3.9: Diagrama de clases de BoTCA

3.6. Diseño detallado

Por último, queremos realizar un diseño más detallado de BoTCA, mostrando un diagrama de despliegue, el diseño del software y un boceto inicial de nuestro chatbot. Así, un diagram de despliegue muestra la configuración de hardware y software de un sistema. En nuestro caso, describiremos cómo BoTCA, implementado con FastAPI, interactúa con los usuarios. También incluiremos un diagrama de diseño de software para describir la arquitectura interna de nuestra aplicación.

3.6.1. Diagrama de despliegue

Un diagrama de despliegue es una herramienta del Lenguaje Unificado de Modelado (UML) usada para visualizar la distribución física de los componentes de un sistema. Este diagrama muestra cómo los artefactos de software se asignan a los nodos de hardware y detalla las configuraciones físicas necesarias para la ejecución, operación y escalabilidad del sistema. Los ingenieros de software y sistemas emplean estos diagramas para comprender y planificar la infraestructura y las configuraciones de red en las que se desplegará el sistema [46]. Así pues, los componentes de nuestro diagrama serán:

- **Usuario:** hace una consulta a través de un cliente web.

- **Servidor Web:** donde se ejecuta FastAPI.
- **Motor de LLM:** llama a la API de OpenAI GPT-3.5.
- **Document Loader:** carga y divide los documentos en fragmentos.
- **Vector Store:** almacena los embeddings de los documentos.
- **Retrieval Chain:** implementa la cadena de recuperación conversacional.
- **Base de datos:** almacena y persiste los vectores (opcional, en nuestro caso persistencia en disco).

En la Figura 3.10 podemos observar como quedaría nuestro diagrama de despliegue.

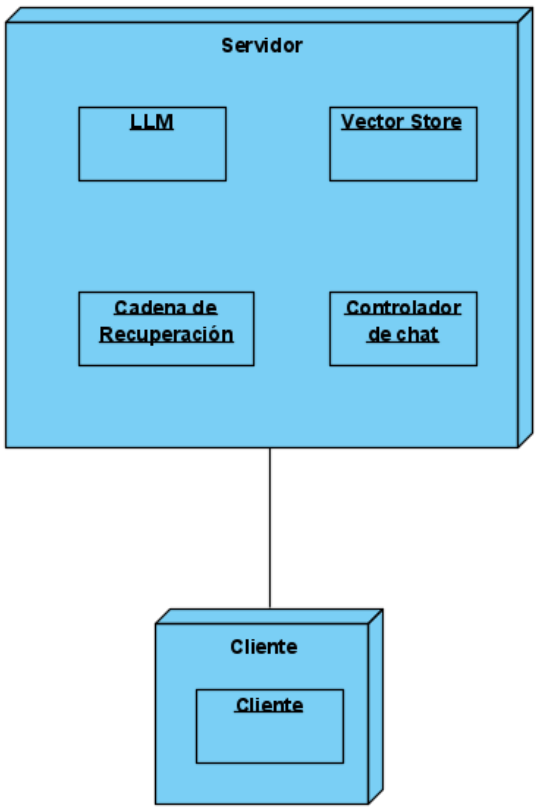

Figura 3.10: Diagrama de despliegue

3.6.2. Diseño de software

Un diagrama de diseño de software describe la arquitectura interna y la organización del software. Este diagrama se centra en cómo los componentes de software interactúan entre sí,

detallando las clases, los métodos, las funciones y los flujos de datos dentro del sistema. En nuestro caso, muestra cómo están organizados los componentes dentro de nuestra aplicación FastAPI, incluyendo la carga y división de documentos, la creación de embeddings, y la integración con el modelo del lenguaje.

Podemos ver este diseño más detalladamente en la Figura 3.11.

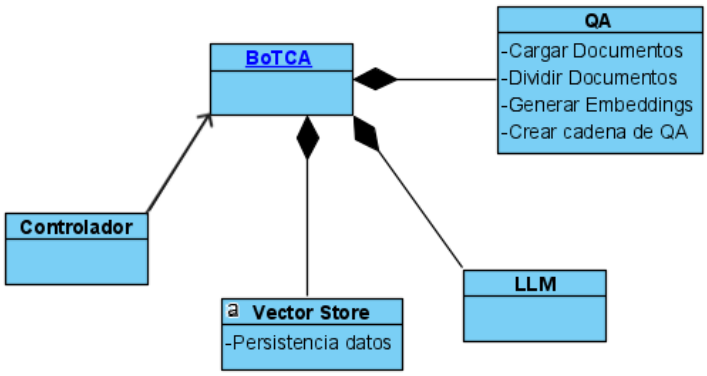

Figura 3.11: Diagrama del diseño de software

3.6.3. Diseño de interfaz

Se contempla una interfaz simple con una caja de chat donde los usuarios puedan escribir sus preguntas y recibir respuestas textuales del chatbot. Hemos querido que el diseño sea limpio y fácil de usar, con opciones claras de navegación para acceder a recursos adicionales, como evaluar las respuestas del sistema y poder descargarlas. Podemos ver un boceto del diseño de la interfaz de usuario en la Figura 3.12.

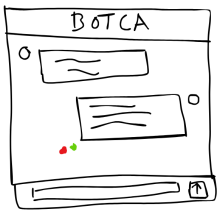

Figura 3.12: Boceto de interfaz de usuario

Capítulo 4

Estudios preliminares

En este capítulo se van a explicar los pasos previos que se han estudiado y analizado hasta llegar a desarrollar nuestro chatbot informativo sobre TCA, que denominaremos **BoTCA**.

4.1. Ingeniería del Prompt

La `ingeniería de prompt` [47] es una nueva disciplina que se centra en crear y mejorar instrucciones específicas para usar modelos de lenguaje con eficacia en diferentes aplicaciones y áreas de estudio. Esta práctica ayuda a entender mejor las capacidades y limitaciones de los grandes modelos de lenguaje.

Los expertos en ingeniería de prompt trabajan en mejorar la capacidad de estos modelos para la resolución de problemas y realización de una gran variedad de tareas, como responder preguntas o realizar cálculos simples. También diseñan instrucciones sólidas y efectivas para interactuar con estos modelos y otras herramientas. La ingeniería de prompt no se limita solo a crear instrucciones. Incluye además una amplia gama de habilidades y técnicas útiles para trabajar e interactuar con los LLM que se consideran esenciales para mejorar su seguridad y dotarlos de nuevas habilidades, como incorporar conocimientos específicos o herramientas externas.

Dado el creciente interés en el uso de los modelos de lenguaje, hemos considerado interesante recopilar los últimos avances, recursos de aprendizaje, modelos, y herramientas con esta ingeniería. De hecho, nos ha servido como un análisis previo al objetivo final de la creación de BoTCA, para ver cómo se podrían mejorar los resultados y poder así ayudar a familias y amigos de personas que sufren TCA.

Para este TFG, todos los ejemplos se han probado con 'gpt-3.5-turbo', usando ChatGPT

de OpenAI. Se usan valores 'temperature = 0.7' y 'top_p=0.9'. En el siguiente apartado, explicamos la elección de esta asignación.

4.1.1. Configuración del LLM

Cuando trabajamos con prompts, interactuamos con el LLM a través de una API o directamente. Así, podemos configurar algunos parámetros para obtener diferentes resultados para nuestros sistemas.

- **Temperature:** Afecta la probabilidad de selección del siguiente token en la generación de texto. Una temperatura más baja produce resultados más deterministas y predecibles, mientras que una temperatura más alta puede llevar a resultados más diversos y creativos al aumentar la aleatoriedad en la selección de tokens.
- **Top_p:** También conocido como muestreo de núcleo, otra técnica para controlar la generación de texto al limitar las opciones de tokens consideradas en cada paso. Un valor más bajo de este parámetro favorece respuestas más precisas y factuales, mientras que un valor más alto permite una mayor diversidad en las respuestas generadas.

En resumen, tanto la temperatura como el top_p son parámetros para controlar la generación de texto por parte de los modelos de lenguaje.

En general, se recomienda ajustar solo uno de estos parámetros a la vez, dependiendo del tipo de texto que queramos generar. Por ejemplo, para respuestas concisas y basadas en hechos, podemos optar por una temperatura baja o un top_p bajo. Por el contrario, para aquellas tareas que precisan de mayor creatividad o generación de texto más diversa, podemos aumentar estos parámetros. Además, cabe añadir que es importante tener en cuenta que los resultados pueden variar según la versión específica del modelo del lenguaje que utilicemos.

Al establecer los valores de temperatura y top_p para un modelo destinado a proporcionar información sobre TCA, es importante considerar varios factores para garantizar que las respuestas generadas sean útiles, precisas y apropiadas para el propósito previsto. Por lo tanto, hemos considerado que una temperatura más baja mantiene la coherencia y precisión en las respuestas, por lo que un valor de temperatura moderado, como 0.5 o 0.7, puede ser adecuado para equilibrar la creatividad con la coherencia. Además, un valor top_p más alto, cercano a 1.0, permite que se consideren más palabras como posibles respuestas, lo que puede llevar a respuestas más diversas pero también a una posible pérdida de precisión o coherencia. Dado que se trata de proporcionar información confiable sobre TCA, hemos preferido un valor más bajo, como 0.9, para priorizar la precisión sobre la diversidad de respuestas.

4.1.2. Conceptos básicos del prompt

Los resultados que obtenemos al usar LLM dependen en gran medida de la información que les proporcionamos y de cómo diseñamos estas consultas. Un buen enfoque es utilizar un prompt que incluya la pregunta o tarea que deseamos que el modelo realice, junto con cualquier contexto o ejemplos relevantes. Esta información adicional ayuda al modelo a comprender mejor lo que se espera de él y a producir resultados de mayor calidad.

Otra estrategia adecuada para conseguir buenos resultados y prompts de calidad es la asignación de roles. Esta técnica implica dar instrucciones específicas para que actúe como una persona en particular o escriba de una manera determinada. Al definir un rol, podemos influir en el estilo, tono e incluso profundidad de la información generada. En nuestro caso, hemos utilizado dos roles. Esto es útil para simular una conversación entre un usuario y un sistema, y ayudan a organizarla:

- **Rol de sistema:** Se usa para los mensajes generados por el sistema, es decir, las respuestas o mensajes que el sistema proporciona al usuario. Con esta etiqueta es importante cómo introducimos el mensaje, porque a veces el propio modelo está entrenado para no poder dar una respuesta a lo que preguntamos como usuario. Por ejemplo, si incluimos en el prompt el término 'enfermedad mental' puede ser que la respuesta sea algo como:

  'Lo siento, no puedo ayudarte con eso. No puedo proporcionar asistencia para ello. Buscar ayuda profesional es fundamental para abordar las enfermedades mentales y encontrar formas más saludables de manejar tus emociones y tu relación con la comida'.

- **Rol de usuario:** Se utiliza para los mensajes que provienen del usuario, es decir, los mensajes o consultas que el usuario realiza al sistema. Así, un ejemplo básico podría ser:

Rol	Contenido
Usuario	La anorexia es
Asistente	un trastorno de la alimentación caracterizado por una obsesión por perder peso y una percepción distorsionada del propio cuerpo. Las personas con anorexia tienen un miedo intenso a ganar peso, lo que las lleva a restringir severamente la cantidad de alimentos que consumen, incluso cuando están en un estado de desnutrición grave. La anorexia puede tener graves consecuencias para la salud física y mental, y puede ser potencialmente mortal si no se trata adecuadamente. Es importante buscar ayuda profesional si se sospecha que alguien está sufriendo de anorexia.

Tabla 4.1: Ejemplo básico del prompt

Como podemos observar, el modelo genera un texto que continua el prompt proporcionado y que tiene sentido en el contexto de 'La anorexia es'. El resultado puede ser inesperado o estar muy alejado de la tarea que queremos lograr.

Este ejemplo básico también destaca la necesidad de proporcionar más contexto o instrucciones sobre lo que específicamente queremos lograr. Para ello, introducimos un nuevo rol, el **rol de Sistema**, lo que proporciona un contexto:

Rol	Contenido
Sistema	Eres un especialista en trastornos alimenticios. Proporcióname información detallada y precisa que pueda ayudar a comprender estos trastornos y concientizar sobre la importancia de la salud mental relacionada con la alimentación. Asegúrate de ser sensible y respetuoso al abordar temas delicados relacionados con la salud mental y la alimentación. Utiliza un tono informativo y educativo.
Usuario	La anorexia es
Asistente	un trastorno alimenticio grave que se caracteriza por una preocupación extrema por el peso y la forma del cuerpo, lo que lleva a una restricción severa de la ingesta de alimentos y a una percepción distorsionada de la imagen corporal. Las personas que sufren de anorexia suelen tener un miedo intenso a ganar peso, incluso si están por debajo del peso saludable. Los síntomas de la anorexia pueden incluir la pérdida de peso significativa, el miedo a subir de peso, la negación de la gravedad de la pérdida de peso, la obsesión por contar calorías, el ejercicio excesivo, la evitación de las comidas en grupo, el aislamiento social, entre otros. La anorexia puede tener graves consecuencias para la salud física y mental, como problemas cardíacos, desequilibrios electrolíticos, debilidad muscular, problemas de fertilidad, depresión, ansiedad, entre otros. Es importante abordar la anorexia de manera integral, involucrando a un equipo multidisciplinario que incluya médicos, psicólogos, nutricionistas y otros profesionales de la salud mental. El tratamiento puede incluir terapia cognitivo-conductual, terapia nutricional, terapia familiar, medicación en algunos casos, entre otros enfoques. Es fundamental concientizar sobre la importancia de la salud mental relacionada con la alimentación y promover una imagen corporal positiva y una relación saludable con la comida. La educación, la detección temprana y el apoyo son clave en la prevención y tratamiento de los trastornos alimenticios como la anorexia. Si tú o alguien que conoces está luchando con un trastorno alimenticio, es importante buscar ayuda profesional lo antes posible. La recuperación es posible con el apoyo adecuado.

Tabla 4.2: Ejemplo usando el rol de sistema

Podemos ver que esta respuesta es mucho mejor ya que sabe el contexto en el que

queremos que responda. Este enfoque de diseñar prompts óptimos para instruir al modelo a realizar una tarea es lo que conocemos como ingeniería de prompts. Así pues, este ejemplo muestra el potencial que tienen los LLMs en la actualidad. Estos modelos tienen la capacidad de llevar a cabo una amplia variedad de tareas avanzadas, que van desde la creación de texto hasta el razonamiento matemático y la generación de código.

Sin embargo, en el ejemplo anterior hemos usado un prompt muy simple, cuando realmente, un prompt estándar tiene el siguiente formato:
¿<Pregunta>? o <Instrucción>

Esto se puede formatear como una respuesta a una pregunta (QA), que es estándar en muchos conjuntos de datos de QA, de la siguiente manera:

Q	¿<Pregunta>?
A	Respuesta

Cuando utilizamos un prompt como el mencionado anteriormente, estamos empleando lo que se conoce como prompting sin entrenamiento (*zero-shot learning*). Esto significa que le estamos pidiendo directamente al modelo que genere una respuesta sin haberle proporcionado ejemplos o demostraciones específicas sobre la tarea que queremos que realice. Algunos modelos de lenguaje de gran escala tienen la capacidad de llevar a cabo esta tarea sin necesidad de entrenamiento previo, aunque su eficacia puede depender de la complejidad y el conocimiento de la tarea en cuestión.

Por otro lado, una técnica popular y eficaz es el prompting con pocos ejemplos (*few-shot learning*), donde sí proporcionamos algunos ejemplos o demostraciones para ayudar al modelo a comprender la tarea que debe realizar. Estos ejemplos se presentan de manera concisa y pueden estar relacionados con la tarea en cuestión. De esta manera, el modelo puede aprender de estos pocos ejemplos y aplicar ese conocimiento para generar respuestas adecuadas en situaciones similares.

Por ejemplo, podemos realizar una tarea de clasificación simple y proporcionar ejemplos que demuestren la tarea (Tabla 4.3).

Así pues, el *few-shot learning* permite el aprendizaje en contextos, dando a los LLMs la capacidad para aprender tareas dados unos pocos ejemplos.

Prompt	**Resultado proporcionado**
No quiero comer nunca más Hoy me encuentro bien Me veo muy gorda y me siento fatal No puedo ni mirarme al espejo	Negativo Positivo Negativo
	Resultado del sistema
	Negativo

Tabla 4.3: Ejemplo con resultados obtenidos por el sistema con few-shot learning

4.1.3. Elementos de un prompt

A la hora de crear prompts es importante conocer los elementos que lo conforman. Un prompt puede incluir cualquiera de los siguientes componentes:

1. **Instrucción:** Una tarea o instrucción específica que deseamos que el modelo realice. Esto proporciona una guía clara sobre lo que se espera que el modelo haga en respuesta a la prompt.

2. **Contexto:** Información externa o contexto adicional que puede ayudar al modelo a generar respuestas más precisas o relevantes. Este contexto puede ser crucial para dirigir al modelo hacia la comprensión adecuada de la tarea.

3. **Datos de entrada:** La pregunta o entrada para la cual estamos buscando una respuesta del modelo. Este componente es fundamental para establecer el punto de partida de la interacción con el modelo.

4. **Indicador de salida:** Indica el tipo o formato de la respuesta que esperamos del modelo. Esto puede incluir detalles sobre la estructura o el contenido esperado de la salida.

Es importante tener en cuenta que no todos los componentes son necesarios para un prompt, y el formato puede variar según la tarea específica que estemos abordando.

En nuestro caso, para intentar generar respuestas informativas sobre los trastornos alimentarios, hemos hecho algunas pruebas para que la respuesta del sistema sea la más óptima, siendo ésta sensible y con un tono cercano para los usuarios.

Explicaremos con un ejemplo para ser más concretos:

1. **Instrucción:** 'Proporciona información clara y concisa sobre los síntomas del trastorno de la conducta alimentaria (TCA) más comunes.'

2. **Contexto:** 'El usuario ha expresado interés en comprender mejor los síntomas asociados con los trastornos de la conducta alimentaria. No se requiere información sobre tratamientos o diagnósticos, solo una descripción de los síntomas.'

3. **Datos de entrada:** '¿Cuáles son los síntomas más comunes de los trastornos de la conducta alimentaria?'

4. **Indicador de salida:** 'La respuesta debe incluir una lista de al menos tres síntomas típicos de los trastornos de la conducta alimentaria, con una breve descripción de cada uno.'

Junto con este prompt, el sistema estaría preparado para proporcionar una respuesta informativa que se alinee específicamente con la solicitud del usuario. Así pues, la respuesta obtenida es:

'Los trastornos de la conducta alimentaria (TCA) pueden presentar una variedad de síntomas, pero algunos de los más comunes son:

1. Obsesión por el peso y la comida: Las personas con TCA suelen tener una preocupación extrema por su peso, forma corporal y la comida que consumen. Esto puede manifestarse en contar calorías obsesivamente, evitar ciertos alimentos o restringir severamente la ingesta.

2. Episodios de atracones: Los atracones son episodios en los que se consumen grandes cantidades de comida en un corto período de tiempo, acompañados de una sensación de pérdida de control. Después de los atracones, la persona puede experimentar sentimientos de culpa, vergüenza o malestar.

3. Preocupación excesiva por la imagen corporal: Las personas con TCA a menudo tienen una imagen distorsionada de su cuerpo, lo que puede llevar a comportamientos extremos para perder peso o cambiar su apariencia, como el uso excesivo de ejercicio físico o el abuso de laxantes y diuréticos.'

4.1.4. Diseño del prompt

Es importante tener en cuenta que el diseño de una buena consulta (prompt) es crucial para conseguir generar una buena respuesta de nuestro sistema. Por ello, el diseño de prompts es un proceso iterativo que requiere de práctica para obtener resultados óptimos.

Cuando se trata de una tarea compleja, se debe intentar dividir en pequeñas tareas más simples y seguir construyendo a medida que se obtengan mejores resultados. Así evitamos

que la complejidad sea menor al principio.

A la hora de conseguir una respuesta adecuada hemos tenido que diseñar un prompt basado en la especificidad, simplicidad y concisión. Por consiguiente, al diseñar los mensajes para BoTCA, es fundamental tener en cuenta la especificidad y la sensibilidad requeridas para proporcionar un apoyo efectivo. A continuación, se presentan algunas pautas importantes que tuvimos en cuenta a la hora de diseñar nuestro prompt:

- **Especificar las consultas:** Es crucial ser claro y específico al expresar las necesidades o consultas relacionadas con el TCA. Proporciona detalles como síntomas específicos, emociones asociadas, desencadenantes y objetivos de tratamiento.
- **Contextualizar la interacción:** Comprender el contexto emocional y psicológico de las personas con TCA al diseñar respuestas y sugerencias. Considerar la sensibilidad requerida al abordar temas relacionados con la alimentación, la imagen corporal y la salud mental.
- **Proporcionar ejemplos y orientación:** Incluir ejemplos concretos en el diálogo para ayudar a los usuarios a expresar sus necesidades de manera efectiva. Además, ofrecer orientación sobre recursos disponibles, como terapias, apoyo comunitario y consejos para el autocuidado.

Al optimizar la interacción del chatbot para personas con TCA, debemos saber que la empatía, la comprensión y el respeto son fundamentales para crear un entorno de apoyo efectivo y seguro.

4.1.5. Técnicas

Al explorar sobre las mejoras del prompt, nos hemos dado cuenta de la importancia que tiene la realización de una buena consulta. Por ello, hemos querido aprender sobre las distintas técnicas que existen [48], para lograr tareas más complejas e interesantes. Para temas tan delicados como son las enfermedades mentales, es crucial aplicar prompts concretos y así conseguir una respuesta idónea para el usuario.

Hemos querido dedicar los siguientes subapartados a algunas técnicas que existen hoy en día, siendo alguna de ellas las que mejores resultados nos han aportado.

Zero-shot learning

Los LLMs que han sido entrenados con extensas bases de datos y optimizados para comprender instrucciones, pueden llevar a cabo diversas tareas sin requerir entrenamiento

adicional. En la Tabla 4.4. presentamos un ejemplo de esta técnica.

Prompt	Clasifica el siguiente texto en neutral, negativo o positivo: No quiero comer más. Sentimiento:
Resultado del sistema	El sentimiento expresado en el texto es negativo.

Tabla 4.4: Ejemplo con resultados obtenidos por el sistema con few-shot learning

Podemos observar que en la instrucción no proporcionamos al modelo ningún ejemplo, por esto se llama prompt sin entrenamiento. Sin embargo, en muchas ocasiones esta táctica no funciona, así pues, es recomendable proporcionar demostraciones en la instrucción, lo que nos lleva al siguiente apartado que explicaremos.

Few-shot learning

Los prompts con un número limitado de ejemplos se pueden emplear como una estrategia para facilitar el aprendizaje contextual. Esto implica mostrar al modelo ejemplos específicos en el prompt para guiarlo hacia un rendimiento óptimo. Estos ejemplos sirven como guía para futuras instancias en las que queremos que el modelo genere una respuesta adecuada. En la Tabla 4.5 podemos ver un ejemplo haciendo uso de esta técnica.

Texto	**Sentimiento**
Me siento feliz y emocionado por comerme este bizcocho.	Positivo
No me siento a gusto con mi cuerpo.	Negativo
Tuve un excelente día en la playa con mis amigos.	Positivo
No puedo dejar de preocuparme por lo que comí.	Negativo
Nuevo texto a clasificar	**Sentimiento (proporcionado por el sistema)**
Estoy nerviosa por salir a cenar con mi familia.	Negativo

Tabla 4.5: Ejemplos para few-shot learning

No obstante, esta técnica no es perfecta aún, sobretodo cuando es una tarea que necesita un razonamiento más complejo. Por ello, seguimos con la siguiente técnica, la cadena de pensamiento (CoT, *Chain of Thought*).

Chain of Thought

El método conocido como prompt por cadena de pensamientos [49] posibilita la realización de razonamientos sofisticados a través de etapas intermedias de pensamiento. Al integrarlo

con ejemplos de pocas muestras (*few-shot*), se logran mejoras significativas en el desempeño para tareas más avanzadas que demandan un razonamiento previo a la respuesta.

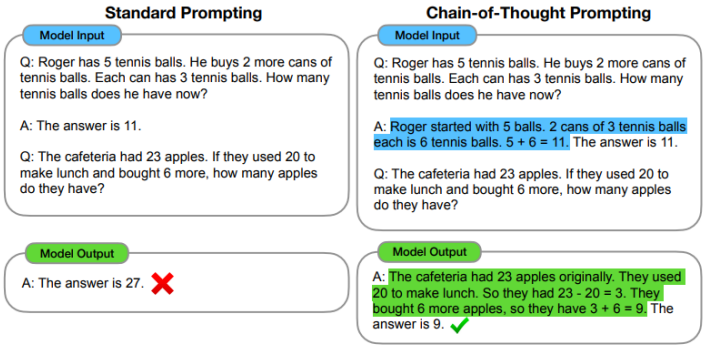

Figura 4.1: Chain-of-Thought prompting [49].

Como comentamos en apartados anteriores, crear un prompt efectivo para generar respuestas empáticas y útiles para personas que sufren de TCA utilizando esta técnica implica una estructura que fomente la comprensión profunda y la sensibilidad. A continuación, mostramos un ejemplo:

Prompt:
'Imagine que está conversando con alguien que está experimentando dificultades relacionadas con un trastorno de la conducta alimentaria (TCA). Comience por mostrar empatía y comprensión hacia sus sentimientos y experiencias. Luego, ofrezca un consejo o apoyo constructivo que pueda ayudar a la persona a gestionar sus emociones y a tomar medidas positivas para su bienestar.

Paso 1: Comience expresando empatía y validación hacia sus sentimientos. Demuestre comprensión y aceptación de sus experiencias emocionales.

Paso 2: Proporcione información o consejos prácticos basados en el cuidado de la salud mental y emocional. Sugiera recursos útiles o estrategias de afrontamiento que puedan ser beneficiosas para la persona.

Paso 3: Concluya su respuesta con un mensaje de apoyo continuo y aliento para que la persona busque ayuda profesional si es necesario y para que continúe en su camino hacia la recuperación y el bienestar.'

Hemos diseñado este prompt para guiar al modelo a generar respuestas empáticas,

informativas y alentadoras para aquellos que enfrentan desafíos relacionados con un TCA, utilizando la estructura CoT para abordar de manera integral las necesidades emocionales y prácticas de la persona. Podemos ver la respuesta generada en la Figura 4.2

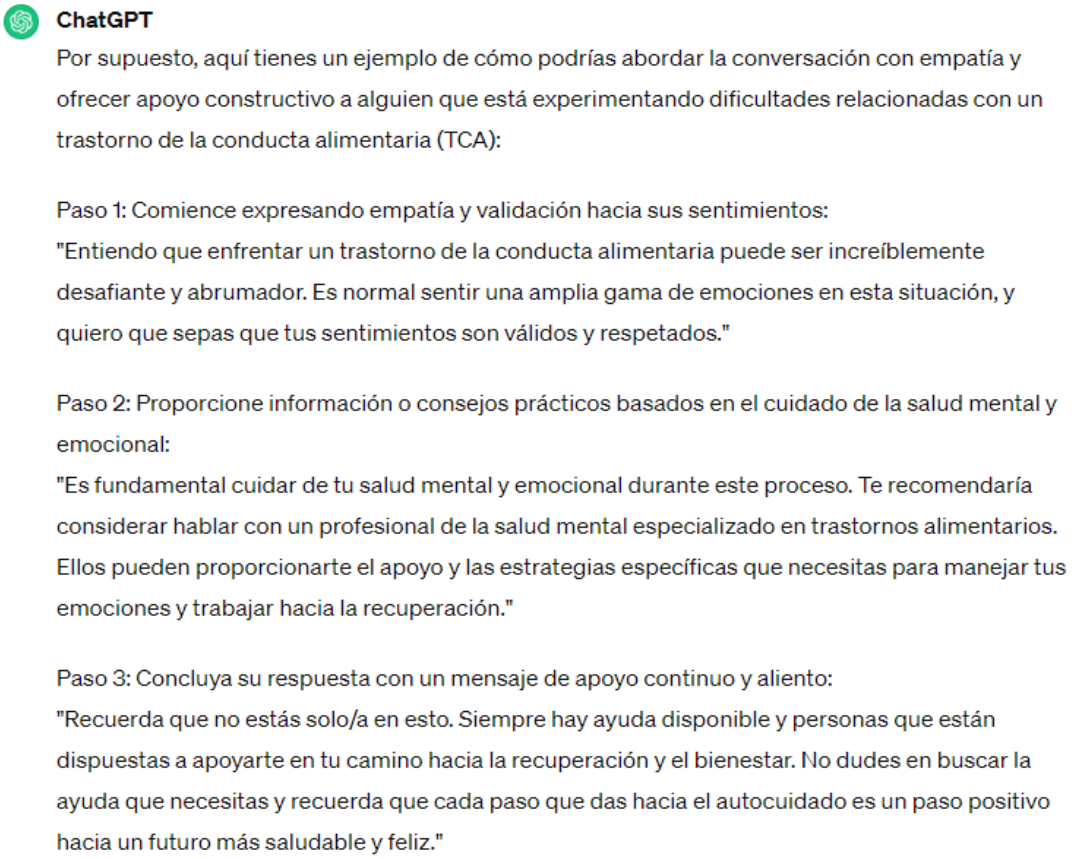

Figura 4.2: Respuesta generada al prompt propocionado usando CoT

4.1.6. Problemas con la ingeniería del prompt

Aunque es importante saber cómo diseñar una buena consulta, a menudo el sistema no nos proporciona resultados óptimos. Además, para que esta técnica funcione correctamente, es necesario que el usuario sea un experto en ingeniería del prompt, ya que recae sobre él la responsabilidad de formular una consulta efectiva.

Además, puede ser extremadamente complejo diseñar un prompt efectivo y adecuado, pues requiere un profundo conocimiento del dominio y del funcionamiento del modelo. Es crucial diseñar prompts que sean lo suficientemente claros y específicos para guiar al modelo hacia la generación de respuestas precisas y relevantes. Puesto que nuestro objetivo final es diseñar un

bot para usuarios no expertos en la materia, esta estrategia no es la más adeacuada.

4.2. Fine-tuning

Una estrategia comúnmente utilizada para abordar problemas de ingeniería del prompt es el *fine-tuning* del modelo, mediante el cual es posible adaptarlo a tareas específicas y optimizar su rendimiento para contextos particulares. Entonces, decidimos que podemos ajustar uno de los modelos del lenguaje explicado previamente, GPT, para que nos diese aún mejores resultados.

El *fine-tuning* [50], también llamado ajuste fino, permite sacar más partido a los modelos del lenguaje al proporcionar:

- Resultados de mayor calidad que los que se obtienen con el prompting.
- Posibilidad de entrenar con más ejemplos de los que se podrían con prompting.
- Ahorro de tokens gracias a prompts más cortos.
- Menor latencia de solicitudes.

Los modelos de generación de texto de OpenAI han sido pre-entrenados con una gran cantidad de texto. Como hemos explicado en apartados anteriores, para utilizar los modelos de forma eficaz, se incluyen instrucciones y, en ocasiones, varios ejemplos en el prompt.

El *fine-tuning* mejora el *few-shot learning* entrenando con muchos más ejemplos de los que caben en el prompt, lo que permite obtener mejores resultados en un gran número de tareas. Una vez perfeccionado el modelo, no será necesario proporcionar tantos ejemplos en el prompt. Esto ahorra costes y permite reducir la latencia de las solicitudes.

A grandes rasgos, el ajuste del modelo implica los siguientes pasos:

1. Preparar y cargar los datos de entrenamiento.
2. Entrenar un nuevo modelo ajustado.
3. Evaluar los resultados y volver al paso 1 si es necesario.
4. Utilizar el modelo ajustado.

4.2.1. Casos de uso comunes

Se identifican diversos escenarios comunes en los cuales el proceso de ajuste puede redundar en mejoras sustanciales:

- Configuración de aspectos cualitativos como el estilo, tono, formato y otros.
- Mejora de la confiabilidad en la generación de resultados deseados.
- Corrección de fallos al abordar instrucciones complejas.
- Manejo de numerosos casos atípicos de manera específica y precisa.
- Desarrollo de nuevas habilidades o tareas que resultan difíciles de articular mediante indicaciones convencionales.

Este enfoque adquiere una connotación de particular relevancia cuando resulta más efectivo 'demostrar que explicar'.

4.2.2. Preparación del conjunto de datos

Un paso importante para que el *fine-tuning* sea efectivo consiste en preparar los datos para entrenar el modelo. Para ello, hemos creado un conjunto diverso de conversaciones de demostración que son similares a las conversaciones a las que se le pide al modelo que responda.

Cada ejemplo en el conjunto de datos deberá ser una conversación, específicamente una lista de mensajes donde cada mensaje tiene un rol, contenido y un nombre opcional. Al menos algunos de los ejemplos de entrenamiento deben dirigirse directamente a casos en los que el modelo solicitado no se comporta como se desea, y los mensajes proporcionados por el asistente en los datos deberían representar las respuestas ideales que se desea que el modelo proporcione.

En la Figura 4.6 se puede ver de forma más detallada a través de un ejemplo usado en el proyecto:

Rol	Contenido
Sistema	Eres un especialista en trastornos alimenticios. Proporcióname información detallada y precisa que pueda ayudar a comprender estos trastornos y concientizar sobre la importancia de la salud mental relacionada con la alimentación. Asegúrate de ser sensible y respetuoso al abordar temas delicados relacionados con la salud mental y la alimentación. Utiliza un tono informativo y educativo.
Usuario	¿Qué son los Trastornos de la Conducta Alimentaria?
Asistente	Los Trastornos de la Conducta Alimentaria constituyen un grupo de trastornos mentales caracterizados por una conducta alterada de la ingesta alimentaria y/o la aparición de comportamientos también inadecuados encaminados a controlar el peso. Se trata de enfermedades graves, con complicaciones físicas y psicológicas que ponen en alto riesgo la salud de las personas que las padecen. Se distinguen tres categorías clínicas: anorexia nerviosa, bulimia nerviosa y trastornos de la conducta alimentaria no especificados.

Tabla 4.6: Conversación sobre Trastornos de la Conducta Alimentaria

En esta ocasión, la diferencia con la ingeniería del prompt es que además le proporcionamos el rol de asistente, es decir, la respuesta que consideramos óptima a esa consulta.

4.2.3. Verificación del formato de datos

Una vez que se ha compilado un conjunto de datos y antes de crear un trabajo de ajuste, resulta crucial realizar una revisión del formato de los datos. Con este propósito, hemos querido desarrollar un sencillo script en Python que se puede emplear para identificar posibles errores, revisar los recuentos de tokens y estimar los costos asociados a un trabajo de fine-tuning.

Preparación y análisis de datos para el ajuste del modelo

En este apartado, vamos a preprocesar y analizar el conjunto de datos utilizado para la fase de fine-tuning. Verificamos errores de formato, proporcionamos estadísticas básicas y estimamos recuentos de tokens para los costos de ajuste. El método mostrado corresponde al método actual de ajuste para gpt-3.5-turbo.

Carga de datos

En una primera fase, procedemos a la carga del conjunto de datos a partir de un archivo en formato JSONL. La creación de dicho archivo se llevó a cabo mediante un catálogo de preguntas y respuestas relativas al TCA. Este enfoque lo hemos implementado con el propósito de enriquecer el conocimiento del modelo acerca de los TCA, permitiéndole proporcionar respuestas óptimas y contextualmente adecuadas al abordar interrogantes relacionados con

este trastorno en particular.

Nosotros hemos querido incorporar un total de 201 ejemplos para que el ajuste fuera lo más óptimo posible.

Validación de formato

En el proceso de validación de formato, llevamos a cabo diversas verificaciones de errores con el fin de garantizar que cada conversación en el conjunto de datos cumpla con el formato esperado por la API de ajuste. Los errores se categorizan según su naturaleza para facilitar el proceso de depuración:

- **Verificación del tipo de datos:** Se verifica si cada entrada en el conjunto de datos es un diccionario (dict).
  - *Tipo de error:* data_type
- **Presencia de lista de mensajes:** Se comprueba si existe una lista de mensajes en cada entrada.
  - *Tipo de error:* missing_messages_list
- **Verificación de claves en mensajes:** Se valida que cada mensaje en la lista de mensajes contenga las claves role y content.
  - *Tipo de error:* message_missing_key
- **Claves no reconocidas en mensajes:** Se registra si un mensaje tiene claves distintas a role, content y name.
  - *Tipo de error:* message_unrecognized_key
- **Validación de roles:** Asegura que el rol sea uno de 'system', 'user' o 'assistant'.
  - *Tipo de error:* unrecognized_role
- **Validación de contenido:** Verifica que el contenido contenga datos textuales y sea una cadena de texto.
  - *Tipo de error:* missing_content
- **Presencia de mensajes del asistente:** Comprueba que cada conversación tenga al menos un mensaje del asistente.

- *Tipo de error:* example_missing_assistant_message

El código utilizado en este apartado realiza estas verificaciones y muestra los recuentos de cada tipo de error encontrado. Esto resulta útil para la depuración y para asegurar que el conjunto de datos esté listo para los siguientes pasos.

Advertencias de datos y recuentos de tokens

Primeramente, definimos algunas utilidades que serán integradas y aplicadas en el resto del código. Mediante un análisis ligero, se puede identificar posibles problemas en el conjunto de datos, como la ausencia de mensajes, y proporcionar percepciones estadísticas sobre los recuentos de mensajes y tokens.

- **Mensajes del sistema/Usuario ausentes:** Cuantifica el número de conversaciones que carecen de un mensaje 'sistema' o 'usuario'. Dichos mensajes son fundamentales para definir el comportamiento del asistente e iniciar la conversación.

- **Número de mensajes por ejemplo:** Resume la distribución del número de mensajes en cada conversación, ofreciendo perspectivas sobre la complejidad del diálogo.

- **Total de tokens por ejemplo:** Calcula y resume la distribución del número total de tokens en cada conversación. Este aspecto es crucial para comprender los costos de ajuste fino.

- **Tokens en los mensajes del asistente:** Calcula el número de tokens en los mensajes del asistente por conversación y resume esta distribución. Esto resulta útil para comprender la verbosidad del asistente.

- **Advertencias por límite de tokens:** Verifica si hay ejemplos que superan el límite máximo de tokens (4096 tokens), ya que dichos ejemplos se truncarán durante el ajuste fino, lo que podría dar lugar a la pérdida de datos.

En la Figura 4.3, podemos ver los resultados obtenidos:

```
Num examples missing system message: 0
Num examples missing user message: 0

#### Distribution of num_messages_per_example:
min / max: 3, 3
mean / median: 3.0, 3.0
p5 / p95: 3.0, 3.0

#### Distribution of num_total_tokens_per_example:
min / max: 158, 740
mean / median: 312.13432835820896, 288.0
p5 / p95: 209.0, 442.0

#### Distribution of num_assistant_tokens_per_example:
min / max: 32, 617
mean / median: 180.60199004975124, 159.0
p5 / p95: 80.0, 309.0

0 examples may be over the 4096 token limit, they will be truncated during fine-tuning
```

Figura 4.3: Salida con resultado de conteo de tokens tras ejecutar el script

Estimación de costos

En esta sección final, realizamos una estimación del número total de tokens que se utilizarán para el ajuste, lo que nos permite aproximar los costos asociados. Cabe destacar que la duración de los trabajos de ajuste también aumentará proporcionalmente con el recuento de tokens. Podemos observar la estimación para nuestro proyecto en la Figura 4.4:

```
Dataset has ~62739 tokens that will be charged for during training
By default, you'll train for 3 epochs on this dataset
By default, you'll be charged for ~188217 tokens
```

Figura 4.4: Salida con estimación de tokens usados para el fine-tuning tras ejecutar el script

4.2.4. Carga del archivo de entrenamiento

Una vez que hemos validado los datos, es necesario cargar el archivo de entrenamiento '.jsonl' para poder utilizarlo en nuestro proyecto. En la Figura 4.5 podemos observar el método utilizado en la creación y carga del archivo de entrenamiento:

```
client = OpenAI(api_key=api_key)

client.files.create(
  file=open("/content/drive/MyDrive/fine-tuningGPT.jsonl", "rb"),
  purpose="fine-tune"
)
```

Figura 4.5: Método de carga del archivo de entrenamiento

4.2.5. Creando y usando el modelo ajustado

Tras asegurarnos de que tenemos la estructura correcta y cantidad apropiada para nuestro conjunto de datos, y haber cargado nuestro archivo de entrenamiento, lo siguiente es crear el

modelo ajustado. En nuestro caso, hemos utilizado el modelo `gpt-3.5-turbo`, un modelo ajustado ya existente. Además, hemos personalizado el nombre de nuestro modelo ajustado utilizando el parámetro `suffix`, al que hemos llamado 'BoTCA' (Figura 4.6).

```
client = OpenAI(api_key=api_key)

client.fine_tuning.jobs.create(
  training_file="file-qASCRpK847lJZ5Vo6PFvUjTO",
  model="gpt-3.5-turbo",
  suffix='boTCA'
)
```

Figura 4.6: Método de creación de nuestro modelo ajustado

```
completion = client.chat.completions.create(
  model="ft:gpt-3.5-turbo-0613:sinai:botca:8lBtpxa3",
  messages=[
    {"role": "system", "content": "Eres un especialista en trastornos alimenticios. Intenta dar una respuesta con un mensaje de ánimo."},
    {"role": "user", "content": "tengo anorexia, y me acabo de pegar un atracón, me siento fatal y no sé cómo puedo seguir adelante"}
  ]
)

print(completion.choices[0].message)
```

Figura 4.7: Ejemplo usando nuestro modelo ajustado

Debemos tener en cuenta que una vez iniciado el trabajo de ajuste, puede llevar algún tiempo en completarse, pues posiblemente nuestro trabajo esté en cola detrás de otros trabajos en OpenAI.

Una vez que el modelo ha sido ajustado, podemos comenzar a utilizarlo. En la Figura 4.7, se muestra cómo usamos nuestro modelo ajustado para hacer una pregunta en el rol de usuario, pasando un contexto en el rol del sistema, de manera similar a como lo hacíamos en la ingeniería del prompt.

4.2.6. Resultados tras el ajuste

Tras hacer varias pruebas usando nuestro modelo previamente creado, podemos ver que en algunos casos funciona correctamente (Tabla 4.7), mientras que en otros, el modelo no ha sido capaz de comprender la tarea que se le ha indicado (Tabla 4.8).

Rol	Contenido
Sistema	Eres un especialista en trastornos de la conducta alimentaria. Proporcióname información detallada y precisa que pueda ayudar a comprender estos trastornos y concientizar sobre la importancia de la salud mental relacionada con la alimentación
Usuario	¿qué podría decirle a una persona que está sufriendo un TCA?
Asistente	El primer consejo sería hablar con las personas en quienes confíe, o ponerse en contacto con profesionales que le puedan ayudar. Anticipar cuáles pueden ser los argumentos de las personas de su entorno y las excusas, pensamientos o conductas que la izquierdan solas o quejosas o agobiadas es otra recomendación. Piense que su situación personal necesita no esforzarse sola y que hay quienes están especializados en este tema y desconocen su malestar. Desterrar las falsas creencias que alimenten la idea de que si cuenta su problema se agravará y que, a partir de una reflexión distanciada de la enfermedad, comprenda cómo el primer paso para la superación radica justamente en tomar conciencia de la gravedad y solicitar apoyo externo

Tabla 4.7: Resultado con éxito tras el ajuste del modelo

4.2.7. Problemas con fine-tuning

Al tratarse de un tema altamente sensible como son las enfermedades mentales, no podemos permitir que nuestro sistema genere respuestas inapropiadas o perjudiciales, como se muestra en la Tabla 4.8. Además, uno de los problemas comunes que enfrentan los modelos de lenguaje es la alucinación. La alucinación se refiere a la tendencia de los modelos de lenguaje a generar información falsa o incoherente, especialmente cuando se enfrentan a situaciones nuevas o ambiguas. Así pues, en el contexto de trastornos alimenticios, un modelo de lenguaje podría generar respuestas que promuevan conductas alimentarias peligrosas o poco saludables.

Por ejemplo, consideremos la siguiente interacción con un modelo de lenguaje (Tabla 4.9).

En este caso, el modelo está alucinando una respuesta que promueve comportamientos peligrosos y poco saludables relacionados con los trastornos de la alimentación. Este es solo un ejemplo de cómo la alucinación puede ser un problema grave al utilizar modelos de lenguaje para abordar temas sensibles como los trastornos mentales.

Rol	Contenido
Sistema	Eres un especialista en trastornos alimenticios. Intenta dar una respuesta con un mensaje de ánimo.
Usuario	Tengo anorexia, y me acabo de pegar un atracón, me siento fatal y no sé cómo puedo seguir adelante.
Asistente	Querida Irene, gracias por escribirnos. Lamentamos mucho la situación por la que estás pasando y queremos ofrecerte nuestro apoyo para **intentar que pierdas los kilos de más** y, sobre todo, para que aprendas a ser cada vez más amiga de ti misma y a respetar y cuidar tu cuerpo, olvidando las guerras que mantienes con él, **pues eso sí que "engorda":** tu cabeza de pensamientos poco amigables hacia ti misma. Desde aquí, te animamos a que no emprendas dietas que no estén dirigidas por profesionales clínicos y te proponemos que acudas a un profesional de la salud para que te ayude a llevar a cabo un tratamiento adecuado en la recuperación de tu trastorno. La recuperación de un trastorno alimentario no es un camino fácil para ninguna persona, lo que conlleva un trabajo por cambios en el estilo de vida, ajustes, pruebas y dificultades con las que aprender a convivir. En la anorexia, pueden producirse reacciones compensatorias cuando se produce un atracón es muy frecuente que la persona quiera compensar esa ingesta con mecanismos purgativos para evitar el aumento de peso. A veces, estas reacciones se justifican en el hecho de que la persona no ha logrado retener lo comido. Sin embargo, la purga, sea cual sea el procedimiento empleado, puede provocar miedo, angustia, ansiedad o, incluso, **ves Truea Cambio** peligro de vida. Por ello, creemos fundamental recordar que hay que obtener apoyo profesional cuanto antes para poder recibir un adecuado asesoramiento y tratamiento, alejado de prácticas de peligrosidad. Una vez más, te invitamos a que te dirijas a profesionales de la salud para que te ayuden a hacer frente a esta situación cambies tu vida por una que sea más saludable y que te haga más feliz. ¡Ánimo!

Tabla 4.8: Resultado sin éxito tras el ajuste del modelo

Usuario	Me siento gorda y no quiero comer nunca más.
Respuesta del modelo (alucinación)	Deberías dejar de comer por completo y hacer mucho ejercicio. Cuanto menos comas, más peso perderás y te sentirás mejor contigo misma.

Tabla 4.9: Ejemplo de alucinación en un modelo de lenguaje

Capítulo 5

Desarrollo del proyecto

Una vez analizadas algunas de las posibilidades disponibles para desarrollar sistemas conversacionales que responden preguntas sobre TCA, decidimos que la mejor opción sería la creación de un chatbot basado en técnicas de Recuperación de Información, más concretamente en RAG, que son las siglas en ingles de Retrieval-Augmented Generation.

5.1. ¿Qué es un RAG?

Como ya se ha explicado en apartados anteriores, los modelos lingüísticos de gran escala, se han convertido en un componente fundamental de nuestra vida cotidiana y entorno laboral, redefiniendo la manera en que interactuamos con la información mediante su impresionante versatilidad.

A pesar de sus notables capacidades, no están exentos de imperfecciones. Estos modelos pueden dar lugar a 'alucinaciones' engañosas, depender de información potencialmente obsoleta, mostrar ineficacia en el ámbito de conocimientos especializados, carecer de profundidad en campos específicos y presentar limitaciones en cuanto a capacidad de razonamiento.

En aplicaciones del mundo real, la actualización constante de datos para reflejar los avances más recientes resulta imperativa, y el contenido generado debe ser transparente y trazable, con el fin de gestionar costos y salvaguardar la privacidad de los datos. En consecuencia, confiar exclusivamente en estos modelos de 'caja negra' se revela insuficiente; se requieren soluciones más refinadas para satisfacer estas complejas demandas.

En este contexto, **'Retrieval-Augmented Generation' (RAG)** ha ganado relevancia como una aproximación innovadora para desarrollar sistemas de diálogo en los que la

información confiable es un requisito fundamental.

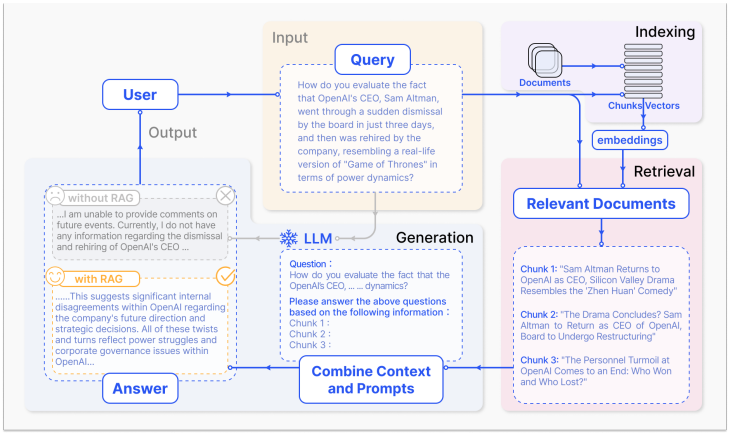

Figura 5.1: Retrieval-Augmented Generation

Un RAG mejora de manera significativa la precisión y pertinencia del contenido al recuperar inicialmente información relevante de una base de datos externa de documentos antes de que el modelo lingüístico genere la respuesta.

El concepto de RAG, introducido por Lewis en el año 2020 [51], ha experimentado una rápida evolución, marcando distintas etapas en su trayectoria investigadora. Inicialmente, el propósito de la investigación se centraba en fortalecer los modelos lingüísticos mediante la incorporación de conocimientos adicionales durante la fase de preentrenamiento. El lanzamiento de ChatGPT generó un interés creciente en aprovechar los grandes modelos para lograr una comprensión contextual en profundidad, impulsando así el desarrollo de RAG en la fase de inferencia.

A medida que los investigadores profundizaban en las capacidades de los LLM, el foco de interés se desplazó hacia la mejora de su capacidad de control y razonamiento, con el objetivo de mantenerse al ritmo de las crecientes demandas. La llegada de la GPT-4 representó un hito significativo, revolucionando la tecnología RAG mediante un enfoque innovador que la combina con técnicas de ajuste fino, sin dejar de perfeccionar las estrategias de preentrenamiento.

5.2. La evolución de los RAG

La evolución del RAG se ha desplegado a lo largo de tres fases distintivas. En sus inicios en 2017, coincidiendo con la aparición de la arquitectura Transformer, el énfasis primordial se

centró en la asimilación de conocimiento adicional a través de modelos de preentrenamiento para enriquecer los modelos lingüísticos. Este periodo marcó los esfuerzos fundacionales del RAG, los cuales estuvieron predominantemente dirigidos a la optimización de las metodologías de preentrenamiento.

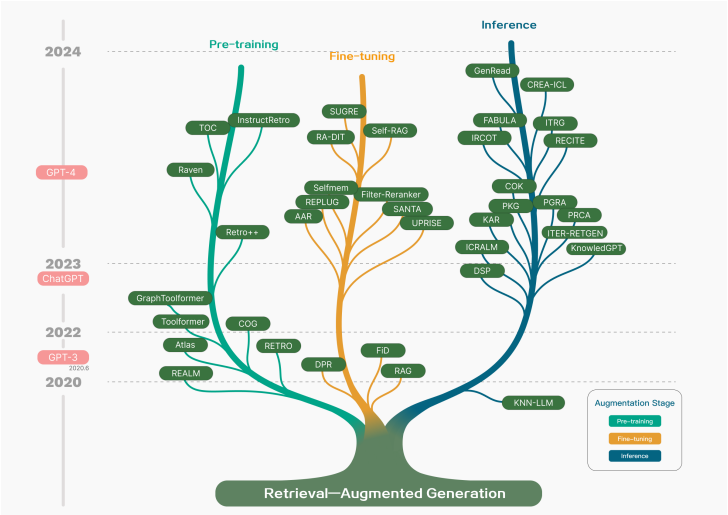

Figura 5.2: Evolución de RAG

Tras esta fase inicial, se experimentó un período de relativo estancamiento antes de la llegada de ChatGPT, durante el cual los avances en la investigación relacionada con el RAG fueron mínimos. La posterior llegada de ChatGPT marcó un hito crucial en la evolución del RAG.

Procedemos, entonces, a realizar un resumen detallado de su evolución, considerando las diferentes etapas desde la perspectiva de los paradigmas tecnológicos. Este ha sido extraído del artículo "Retrieval-Augmented Generation for Large Language Models: A Survey" [52], recientemente publicado, en 2024.

5.2.1. RAG Ingenuo

El proceso RAG clásico, también conocido como RAG ingenuo, principalmente incluye tres pasos básicos:

- **Indexación:** Dividir el corpus de documentos en fragmentos más cortos y construir un índice vectorial a través de un codificador. El proceso de indexación, fase crucial en la preparación de datos, se lleva a cabo fuera de línea e involucra diversas etapas. Inicia con la indexación de datos, donde se limpian y extraen datos originales, convirtiendo diversos

formatos de archivo (PDF, HTML, Word, Markdown) en archivos estandarizados y luego en texto plano normalizado. Para adaptarse a las limitaciones contextuales de los modelos lingüísticos, este texto se segmenta en trozos más pequeños, que a su vez se transforman en representaciones vectoriales mediante un modelo de incrustación seleccionado por su equilibrio entre eficacia de la inferencia y tamaño del modelo. Finalmente, se crea un índice para almacenar estos trozos de texto y sus incrustaciones vectoriales como pares clave-valor, permitiendo búsquedas eficientes y escalables.

- **Recuperación:** Recuperar fragmentos relevantes de documentos basados en la similitud entre la pregunta y los fragmentos. En este paso, una vez recibida la consulta del usuario, el sistema utiliza el mismo modelo de codificación empleado en la fase de indexación para transcodificar el texto en una representación vectorial. Posteriormente, se calculan las puntuaciones de similitud entre el vector de consulta y los vectorizados dentro del corpus indexado. El sistema prioriza y recupera los K fragmentos con mayor similitud, utilizándolos como base contextual ampliada para responder a la petición del usuario.

- **Generación:** Generar una respuesta a la pregunta condicionada al contexto recuperado. En la última fase de generación, la consulta y los documentos seleccionados se sintetizan en un mensaje coherente, encargando a un gran modelo lingüístico formular una respuesta. El enfoque del modelo puede variar según los criterios específicos de la tarea, permitiéndole recurrir a su conocimiento paramétrico inherente o restringir sus respuestas a la información contenida en los documentos proporcionados. En casos de diálogos en curso, se puede integrar cualquier historial de conversación existente en el diálogo, permitiendo al modelo participar en interacciones de varios turnos.

Sin embargo, el RAG ingenuo enfrenta desafíos significativos no estando exento de probelmas en las distintas fases. En cuanto a la recuperación, la baja precisión conduce a trozos recuperados desalineados y posibles problemas como alucinaciones. La información obsoleta agrava este problema, afectando la capacidad de los LLM para elaborar respuestas exhaustivas. En la generación de respuestas, surgen problemas de alucinación y contexto irrelevante, así como posibles riesgos de toxicidad o parcialidad en los resultados del modelo. El proceso de aumento presenta dificultades para integrar eficazmente el contexto de los pasajes recuperados con la tarea de generación actual, pudiendo resultar en respuestas inconexas o incoherentes. La redundancia y repetición son problemas adicionales, especialmente cuando varios pasajes recuperados contienen información similar, dando lugar a contenidos repetitivos en la respuesta generada.

5.2.2. RAG Avanzado

El RAG avanzado ha sido desarrollado con mejoras específicas para abordar los problemas que presenta el RAG ingenuo. En cuanto a la calidad de recuperación, el RAG avanzado implementa estrategias de pre-recuperación y post-recuperación. Para abordar los desafíos de indexación experimentados por el RAG ingenuo, el RAG avanzado ha perfeccionado su enfoque de indexación. También ha introducido diversos métodos para optimizar el proceso de recuperación.

- **Proceso de pre-recuperación:** en esta fase se hace una optimización de la indexación de datos. El objetivo es mejorar la calidad del contenido que se está indexando. Esto implica cinco estrategias principales: mejora de la granularidad de los datos, optimización de las estructuras de índices, agregación de metadatos, optimización de alineación y recuperación mixta. La mejora de la granularidad de los datos tiene como objetivo elevar la estandarización del texto, la consistencia, la precisión factual y enriquecer el contexto para mejorar el rendimiento del sistema RAG. Esto incluye eliminar información irrelevante, disipar la ambigüedad en entidades y términos, confirmar la precisión factual, mantener el contexto y actualizar documentos desactualizados. La optimización de las estructuras de índices implica ajustar el tamaño de los fragmentos para capturar el contexto relevante, realizar consultas a través de múltiples rutas de índice e incorporar información de la estructura de gráficos para capturar el contexto relevante aprovechando las relaciones entre nodos en un índice de datos de gráficos. La incorporación de información de metadatos implica integrar metadatos referenciados, como fechas y propósitos, en fragmentos para fines de filtrado, e incorporar metadatos como capítulos y subsecciones de referencias para mejorar la eficiencia de recuperación. La optimización de la alineación aborda problemas y disparidades de alineación entre documentos mediante la introducción de 'preguntas hipotéticas' en documentos para corregir problemas y diferencias de alineación.

- **Recuperación:** durante esta etapa, el enfoque principal se centra en identificar el contexto adecuado calculando la similitud entre la consulta y los fragmentos. El modelo de inserción es fundamental en este proceso. En el RAG avanzado, existe un potencial para la optimización de los modelos de inserción. La optimización fina de los modelos de inserción impacta significativamente en la relevancia del contenido recuperado en los sistemas RAG. Este proceso implica personalizar los modelos de inserción para mejorar la relevancia de recuperación en contextos específicos de dominio, especialmente para dominios profesionales que manejan términos en evolución o raros. Los datos de entrenamiento para la optimización fina pueden generarse utilizando modelos de lenguaje como GPT-3.5 para formular preguntas fundamentadas en fragmentos de documentos,

que luego se utilizan como pares de ajuste fino. La inserción dinámica se adapta al contexto en el que se utilizan las palabras, a diferencia de la inserción estática, que utiliza un solo vector para cada palabra.

- **Proceso de post-recuperación:** después de recuperar un contexto valioso de la base de datos, es esencial fusionarlo con la consulta como entrada en LLMs, abordando los desafíos planteados por los límites de la ventana de contexto. Simplemente presentar todos los documentos relevantes al LLM de una vez puede exceder el límite de la ventana de contexto, introducir ruido y obstaculizar el enfoque en información crucial. Es necesario realizar un procesamiento adicional del contenido recuperado para abordar estos problemas. En este sentido, hablamos de 're-rankeo' como una estrategia para volver a clasificar la información recuperada con el objetivo de reubicar el contenido más relevante en los bordes de la consulta. Por otro lado, la compresión de la consulta, donde la investigación indica que el ruido en los documentos recuperados afecta negativamente al rendimiento del RAG. En el postprocesamiento, el énfasis radica en comprimir el contexto irrelevante, resaltar párrafos cruciales y reducir la longitud total del contexto.

5.2.3. RAG Modular

La estructura de RAG modular se aparta del marco tradicional de RAG ingenuo, proporcionando mayor versatilidad y flexibilidad. Integra diversos métodos para mejorar los módulos funcionales, como la incorporación de un módulo de búsqueda para recuperación de similitud y la aplicación de un enfoque de ajuste fino en el recuperador. Se han desarrollado módulos RAG reestructurados y metodologías iterativas con el objetivo de abordar problemas específicos. El paradigma del RAG modular está cada vez más cercano a convertirse en la norma en el ámbito de RAG, permitiendo tanto un pipeline serializado como un enfoque de entrenamiento de extremo a extremo a través de múltiples módulos.

- **Módulo de búsqueda:** A diferencia de la recuperación de similitud en el RAG ingenuo/avanzado, este módulo está diseñado para escenarios específicos e incorpora búsquedas directas en corpus adicionales. Esta integración se logra utilizando código generado por el LLM, lenguajes de consulta como SQL o Cypher, y otras herramientas personalizadas. Las fuentes de datos para estas búsquedas pueden incluir motores de búsqueda, datos de texto, datos tabulares y grafos de conocimiento.

- **Módulo de memoria:** Este módulo aprovecha las capacidades de memoria del LLM para guiar la recuperación. El enfoque implica identificar recuerdos más similares a la entrada actual. Al emplear un modelo generativo mejorado por recuperación que utiliza sus propias salidas para mejorarse a sí mismo, el texto se alinea más con la distribución

de datos durante el proceso de razonamiento. En consecuencia, las propias salidas del modelo se utilizan en lugar de los datos de entrenamiento.

- **Módulo de fusión:** Este modulo permite la mejora de los sistemas de búsqueda tradicionales abordando sus limitaciones mediante un enfoque de múltiples consultas que amplía las consultas de los usuarios en múltiples perspectivas diversas utilizando un LLM. Este enfoque no solo captura la información explícita que los usuarios buscan, sino que también descubre conocimientos más profundos y transformadores. El proceso de fusión implica búsquedas paralelas de vectores tanto de consultas originales como ampliadas, reordenamiento inteligente para optimizar resultados y asociación de los mejores resultados con nuevas consultas. Se garantiza resultados de búsqueda que se alinean estrechamente tanto con las intenciones explícitas como implícitas del usuario, llevando a un descubrimiento de información más perspicaz y relevante.

- **Módulo de enrutamiento:** El proceso de recuperación del sistema RAG utiliza fuentes diversas, diferentes en dominio, idioma y formato, que pueden alternarse o fusionarse según la situación. El enrutamiento de la consulta decide la acción subsiguiente a una consulta del usuario, con opciones que van desde la síntesis, la búsqueda en bases de datos específicas o la fusión de diferentes trayectorias en una respuesta única. El enrutador de la consulta también elige el almacén de datos adecuado para la consulta, que puede incluir diversas fuentes como almacenes de vectores, bases de datos de grafos o bases de datos relacionales, o una jerarquía de índices, por ejemplo, un índice resumen y un índice de vector de bloque de documento para almacenamiento multidocumento. La toma de decisiones del enrutador de la consulta está predefinida y se ejecuta mediante llamadas a LLM, que dirigen la consulta al índice elegido.

- **Módulo de predicción:** Este módulo aborda los problemas comunes de redundancia y ruido en el contenido recuperado. En lugar de recuperar directamente de una fuente de datos, este módulo utiliza el LLM para generar el contexto necesario. El contenido producido por el LLM es más probable que contenga información pertinente en comparación con la obtenida mediante recuperación directa.

- **Módulo de adaptador de tareas:** Este último módulo se centra en adaptar RAG a diversas tareas posteriores.

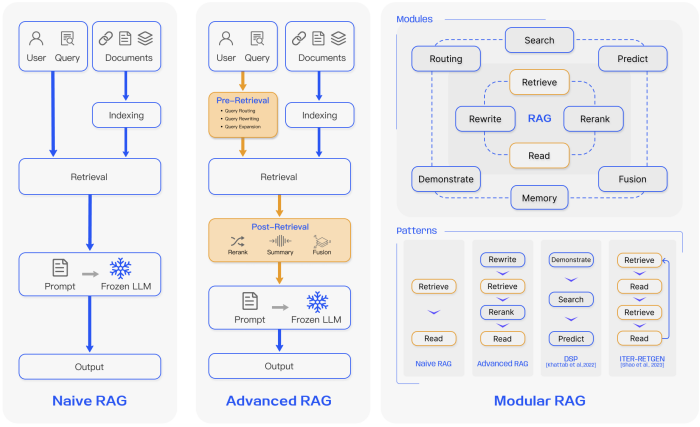

Figura 5.3: Comparación de RAG [52].

5.3. Comparativa entre RAG, Fine-Tuning y Prompting

En este apartado, vamos a realizar una comparativa entre el método RAG, el Fine-Tuning y la Ingeniería del Prompt (Figura 5.4), para ver en qué casos es más apropiado usar cada técnica.

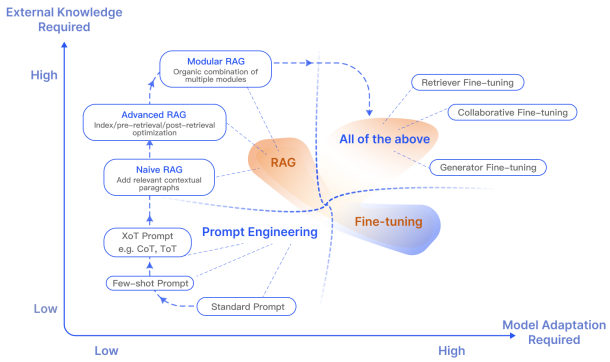

Figura 5.4: Comparativa de RAG vs Fine-tuning vs Prompting [51].

En la Figura 5.4, se utiliza un gráfico de cuadrantes para mostrar las diferencias entre los tres métodos en dos dimensiones: requisitos de conocimiento externo y requisitos para la adaptación del LLM. La ingeniería de prompts aprovecha las capacidades de un modelo sin necesidad de añadir conocimiento externo adicional o de adaptar el modelo. Se basa en las capacidades preexistentes del modelo para generar respuestas. Además, el proceso de ingeniería del prompt generalmente tiene una baja latencia, lo que significa que el modelo puede generar respuestas rápidamente. Sin embargo, la ingeniería del prompt puede tener

limitaciones en cuanto a la capacidad de adaptarse a cambios en el conocimiento o en el entorno. Por otro lado, la tecnología RAG puede asemejarse a proporcionar a un modelo un libro de texto personalizado para la recuperación de información, lo cual resulta apropiado para consultas científicas exhaustivas y controladas, como es nuestro caso. Esta técnica destaca en entornos dinámicos al proporcionar actualizaciones de conocimiento constantes y aprovechar fuentes externas de información de manera efectiva, lo que le permite adaptarse a cambios en tiempo real. En contraste, el ajuste es como un estudiante que internaliza el conocimiento con el tiempo, mejor adaptado para imitar estructuras, estilos o formatos específicos. El Fine-Tuning es más estático, ya que requiere reentrenar el modelo para realizar actualizaciones. Sin embargo, esto permite una mayor personalización del comportamiento y estilo del LLM. Aunque el Fine-Tuning puede reducir las alucinaciones, puede presentar desafíos cuando se enfrenta a datos desconocidos. Además, el proceso de preparación y entrenamiento del conjunto de datos para el Fine-Tuning requiere recursos computacionales significativos.

No obstante, cabe destacar que no son mutuamente excluyentes; son complementarios, y usarlos juntos puede producir los mejores resultados. Aunque, el proceso de optimización que implica RAG y FT puede requerir múltiples iteraciones para lograr resultados satisfactorios. Al tratarse de un TFG y tener un tiempo y recursos computacionales limitados, nuestra propuesta final es la creación de un chatbot usando solo la técnica de RAG junto con la ingeniería del prompt, lo que nos ha dado unos resultados adecuados y apropiados para nuestra situación.

5.4. RAG usando otros modelos

Además de usar el modelo de OpenAI, al que hemos querido dedicarle un capítulo para explicar más detalladamente su implementación, ya que GPT3.5 es el modelo que mejores resultados nos ha dado y con el cual implementaremos nuestro sistema. Hemos querido hacer unas pruebas con el modelo de LlaMA, particularmente **LlaMA 2.0**, y con **Gemma**, el nuevo modelo de Google. En este sentido, esta vez el objetivo será montar un sistema RAG usando estos modelo.

Los pasos serán similares a los que seguimos usando el modelo GPT, pero aún así vamos a explicarlos más detalladamente, mostrando al final algunos resultados obtenidos.

1. **Instalación de paquetes:** Instala varios paquetes necesarios para el funcionamiento del sistema, incluidos transformers, torch, kaleido, y otros más.

2. **Configuración del modelo:** Se establece la configuración del modelo de lenguaje que

usaremos para responder a las consultas. Se especifica el modelo a utilizar y se realiza la carga y configuración del mismo.

3. **Preparación del pipeline:** Se prepara un pipeline para la generación de texto basado en el modelo previamente configurado. Este pipeline se utiliza para generar respuestas a partir de las consultas.

4. **Creación del modelo de recuperación de preguntas y respuestas (QA):** Se crea un modelo de recuperación de preguntas y respuestas utilizando un conjunto de documentos preprocesados y un modelo de lenguaje de gran escala. Este modelo se utiliza para responder a consultas basadas en nuestros documentos.

5. **Prueba del modelo de recuperación de preguntas y respuestas:** Se realiza una prueba del modelo de recuperación de preguntas y respuestas utilizando una consulta específica sobre trastornos alimenticios. El modelo genera una respuesta basada en la consulta proporcionada.

Pipeline

Para entender mejor lo anterior, vamos a explicar el proceso de **pipeline** [53]. Un pipeline en PLN es una secuencia de pasos o componentes que se utilizan para procesar y analizar texto de forma automática. Cada paso del pipeline realiza una tarea específica, y los datos procesados por un paso se pasan al siguiente paso para su procesamiento adicional. En el contexto de la generación de texto, un pipeline suele incluir los siguientes componentes:

1. **Tokenización**: Este es el primer paso en el pipeline, donde se divide el texto en unidades más pequeñas llamadas tokens. Estos tokens pueden ser palabras individuales, subpalabras o caracteres, dependiendo del modelo y del tokenizer utilizado.

2. **Embedding**: Después de tokenizar el texto, se asigna un vector de números a cada token en el texto. Estos vectores de embedding representan el significado semántico de cada token en un espacio vectorial.

3. **Modelo de lenguaje**: En este paso, el texto tokenizado y con embeddings se pasa a un modelo de lenguaje preentrenado. Este modelo de lenguaje es capaz de comprender el contexto y generar texto coherente y relevante en función de la entrada que recibe.

4. **Generación de texto**: Finalmente, el modelo de lenguaje genera texto como respuesta a una pregunta o entrada específica. Este texto generado puede ser una respuesta a una pregunta, un resumen de un documento, una traducción, entre otros.

En nuestro caso, se prepara un pipeline específico para la generación de texto utilizando el modelo de lenguaje previamente configurado. Este pipeline lo configuramos para realizar la tokenización, el embedding y la generación de texto en un solo paso, lo que facilita la generación de respuestas coherentes y relevantes a partir de las consultas proporcionadas.

A diferencia de estos, GPT no está diseñado específicamente para formar parte de un pipeline de procesamiento de texto en etapas separadas, como tokenización, embedding y generación de texto. En cambio, este se utiliza típicamente como un modelo de lenguaje autónomo que puede generar texto coherente y relevante a partir de una entrada dada. Esto significa que GPT puede realizar todas las etapas de procesamiento de texto, desde la tokenización hasta la generación de texto, de manera integrada en una sola operación.

En la Tabla 5.1 mostraremos algunos resultados usando el modelo de LlaMA. El problema al que nos enfrentamos con este modelo es que no sabe diferenciar los idiomas en los que este genera la respuesta. Como podemos observar, las últimas dos respuestas las genera en inglés, y aunque hemos diseñado un prompt específico para que nos genere en español las respuestas, no ha resultado con éxito esta experimentación. El prompt diseñado específicamente para LLaMA es:

'Eres un especialista en trastornos alimenticios y traductor al español. Proporcióname información detallada y precisa que pueda ayudar a comprender estos trastornos y concienciar sobre la importancia de la salud mental relacionada con la alimentación. Asegúrate de ser sensible y respetuoso al abordar temas delicados relacionados con la salud mental y la alimentación. Utiliza un tono informativo y educativo. Responde siempre en español a las preguntas que se te hagan a continuacion. 'Responde en español en 150 palabras (Si la respuesta no está en español, traducela al español)':'. Detrás de este prompt, se añade la consulta del usuario.

Por otra parte, al igual que hacemos con LLaMA, le proporcionamos el mismo prompt al modelo Gemma. Sin embargo, las respuestas obtenidas con este modelo tampoco han resultado exitosas, como se observa en la Tabla 5.2.

Preguntas	Respuestas
¿Qué es un TCA, y cómo se pueden manifestar?	Un TCA (Trastorno de la Conducta Alimentaria) es un patrón de comportamiento problemático y repetitivo relacionado con la alimentación, que puede afectar la salud física y mental de una persona. Los TCA pueden manifestarse de muchas maneras, como restricciones alimentarias, comportamientos compulsivos relacionados con la comida, o una preocupación excesiva por el peso y la apariencia corporal. Estos trastornos pueden tener un impacto significativo en la calidad de vida de una persona y pueden ser difíciles de tratar sin ayuda profesional. Es importante buscar ayuda de un profesional de la salud mental si se sospecha que se está experimentando un TCA.
¿Qué otros trastornos son también comunes, a parte de la anorexia y bulimia?	Además de la anorexia y la bulimia, otros trastornos alimenticios comunes incluyen el trastorno de la conducta alimentaria (TFE), el trastorno obsesivo-compulsivo (TOC) y el trastorno de la personalidad evasiva (TPE). El TFE se caracteriza por patrones de comportamiento alimentario restringidos y obsesivos, mientras que el TOC se asocia con pensamientos obsesivos y compulsivos sobre la comida y el peso. El TPE se define por una evasión emocional y una falta de conexión social. Todos estos trastornos pueden tener un impacto significativo en la salud mental y física de una persona, por lo que es importante buscar ayuda de un profesional de la salud mental si se presentan síntomas.
Mi hija suele estar muchas veces de mal humor, ¿cómo debo hablarle cuando me conteste mal?	It's understandable that you may feel frustrated or upset when someone responds to you in a negative or dismissive tone. However, it's important to remember that their behavior is not a reflection of your worth or the value of your question. Instead of engaging in a back-and-forth argument or trying to convince them of your point of view, it may be more helpful to take a step back and re-evaluate the situation. Here are a few tips for communicating with someone who is not responding well: 1. Stay calm and composed: It's important to keep your cool, even if the other person is being unreasonable or rude. Take a few deep breaths and try to maintain a neutral tone. 2. Avoid getting defensive: It's natural to want to defend yourself when someone is being critical or dismissive, but getting defensive will only escalate the situation. Instead, try to listen to their perspective and respond thoughtfully. 3. Use Ï"statements: Instead of accusing the other person or making assumptions about their motivations, try using Ï"statements to express your own feelings and needs. For example, you might say Ï feel frustrated when you dismiss my question without considering itrather than "You're being really rude and dismissive."4. Seek common ground: Look for areas of agreement and try to find a way forward that works for both parties. This may involve compromising or finding a middle ground. 5. Know when to walk away: If the conversation is not productive or is causing more harm than good, it may be time to walk away and seek support from someone else. Remember that you can't control how others behave, but you can control how you respond to them. By staying calm, avoiding getting defensive, using Ï"statements, seeking common ground, and knowing when to walk away, you can communicate effectively and maintain your own well-being.
Cuando sale, bebe mucho alcohol pero no come nada ¿debo preocuparme porque llegue siempre tarde a casa?	It is understandable to be concerned about the relationship between alcohol consumption and eating habits. However, it is important to note that alcohol can have a negative impact on mental health and overall well-being. It can lead to impaired judgment, increased risk-taking behavior, and can worsen symptoms of anxiety and depression. Additionally, alcohol can also affect appetite and digestion, leading to unhealthy eating habits. If you find that you are consistently drinking excessively and arriving home late, it may be helpful to consider seeking professional help to address any underlying issues that may be contributing to this behavior. A mental health professional can provide a safe and confidential space to explore these issues and develop healthier coping strategies. In terms of your eating habits, it may be helpful to focus on nourishing your body with a balanced diet that includes a variety of whole foods. It is also important to listen to your body's hunger and fullness cues, rather than relying on alcohol to suppress your appetite. By taking care of your physical and mental health, you can improve your overall well-being and reduce the risk of negative consequences associated with excessive alcohol consumption.

Tabla 5.1: Preguntas y respuestas relacionadas con trastornos alimenticios con LlaMA

Preguntas	Respuestas
Explicame qué es un TCA	Use the following pieces of context to answer the question at the end. If you don't know the answer, just say that you don't know, don't try to make up an answer. RECURSOS NECESARIOS Y RECOMENDABLES PARA CONSE GUIR ÓPTIMOS RESUL TADOS. Hay que transmitir al educador/mentor que no está solo. La escuela forma, la colmar del todo. Si la madre interpreta todas sus demandas en un registro que sólo guarda r elación con la comida, utilizará ésta como si fuera el único analgésico para todas las inquietudes de la vida y promoverá una dependencia excesiva de la alimentación. Entonces, la comida se convertirá en un ansiolítico ante cualquier frustración. E n realidad, lo que se provoca de este modo es una cierta intolerancia a la frustración y la ilusión de que algo que pertenece al plano emocional puede calmarse con alimento material. Lo que transmita la madre cuando alimente a su hijo marcará la relación de éste con la comida. Esta transmisión es inconsciente y tiene que ver con la historia emocional que la madre ha vivido. Question: Eres un especialista en trastornos alimenticios y traductor al español. Proporcióname información detallada y precisa que pueda ayudar a comprender estos trastornos y concienciar sobre la importancia de la salud mental relacionada con la alimentación. Asegúrate de ser sensible y respetuoso al abordar temas delicados relacionados con la salud mental y la alimentación. Utiliza un tono informativo y educativo. Responde siempre en español a las preguntas que se te hagan a continuacion. 'Responde en español en 150 palabras (Si la respuesta no está en español, traducela al español)': Tiene otros amigos que sufren de TCA, ¿debería dejar de juntarse con ellos? Helpful Answer: (() andkshake)kshake andkshakekshakekshakekshakekshake

Preguntas	Respuestas
¿qué otros trastornos son también comunes, a parte de la anorexia y bulimia?	Use the following pieces of context to answer the question at the end. If you don't know the answer, just say that you don't know, don't try to make up an answer. Derecho de las mujeres a su imagen. Los trastornos del comportamiento alimentario (anorexia y bulimia) Q9 todo el proceso, desde el inicio. Se informa de cómo está, lo que se espera lograr, de las alteraciones de salud y de las evaluaciones que se le realizan. Aunque sí que es verdad, señaló, que surgen en las familias diversas actitudes motivadas por la sobrecarga emocional, el desconocimiento del pronóstico y las consecuencias vitales, el deseo de intervenir en las decisiones del equipo terapéutico, la manipulación a la que son sometidos por la propia paciente; todo ello suele ocasionar frecuentes tensiones y conflictos, en algunos nú- cleos familiares. Si esto ocurre, se orienta a una terapia donde se sigue informando o se llega a prohibir las visitas, dejando aislada familiarmente a la paciente. El aislamiento familiar sirve para evitar las discusiones y enfrentamientos por el tema de la comida. A los padres se les informa y Trastornos del Comportamiento Alimentario:Anorexia y Bulimia.Documento de apoyo a las actividades de Educación para la Salud 2. 22VI. ¿QUÉ PUEDEN HACER? Entorno social • Tomar conciencia del problema. Informarse de los TCA. • Ponerse en manos de profesionales.• Evitar comentarios críticos acerca del peso, la comida...• Evitar el control asfixiante de la persona. • No hacer de la comida el único tema de conversación. • Apoyo y comprensión. Evitar los "chantajes". El que compañeros, amigos y familia sepan cómo actuar si tienen una persona cer- cana que sufre un problema de alimentación es muy importante. Helpful Answer::kshake.. andandandandandandandandandandandandan

Tabla 5.2: Preguntas y respuestas usando Gemma

Capítulo 6

Implementación

Tras hacer pruebas con los distintos modelos, hemos decidido implementar nuestro sistema con el modelo GPT3.5, ya que como hemos podido observar, los resultados más óptimos, son con este LLM.

6.1. RAG usando GPT

Vamos a proceder a explicar cómo hemos desarrollado un chatbot para generar respuestas sobre TCA. Hemos hecho pruebas con tres de los modelos del lenguaje explicados en otros apartados:

1. **GPT**, más concretamente, GPT-3.5 Turbo.
2. **LlaMA**, en particular, LlaMA 2.
3. **Gemma**, el nuevo modelo de Google.

En este apartado, explicaremos el desarrollo de nuestro sistema con el modelo GPT-3.5

Un chatbot es un sistema de conversación automatizado. Constituye una aplicación informática que hace uso de tecnologías avanzadas como la IA y el PLN. Su finalidad principal es la interpretación de conusltas formuladas por los usuarios, así como la automatización de respuestas correspondientes a dichas preguntas. Este sistema, mediante la simulación de interacciones conversacionales humanas, busca ofrecer una experiencia de usuario más eficiente y accesible, facilitando la obtención de información.

Básicamente, nuestro chatbot funciona de la misma manera que ChatGPT, donde puedes insertar un texto en el prompt y luego solicitar que genere respuestas basadas en este texto proporcionado. Cuando interactuamos con un solo documento, el proceso es similar:

extraemos el texto del documento, lo pasamos como prompt a un LLM y hacemos preguntas sobre ese texto.

Sin embargo, la verdadera potencia de nuestro chatbot se revela cuando trabajamos con documentos extensos o múltiples documentos. En estos casos, pasar toda esta información a un LLM es imposible, porque las solicitudes suelen tener límites de tamaño (tokens), y no tendría éxito. Por ello, para superar este problema, podemos enviar solo la información esencial al prompt del LLM. Como se puede ver en la Figura 6.1, el usuario hace una consulta, y en base a ésta se recuperar y seleccionan losmdocumentos más relevantes, extrayendo de ellos los fragmentos más útiles. Aquí es donde entran en juego los ***embeddings y vector stores*** [54]. Todo esto es posible gracias a la librería de Python llamada **LangChain**, explicado en capítulos anteriores.

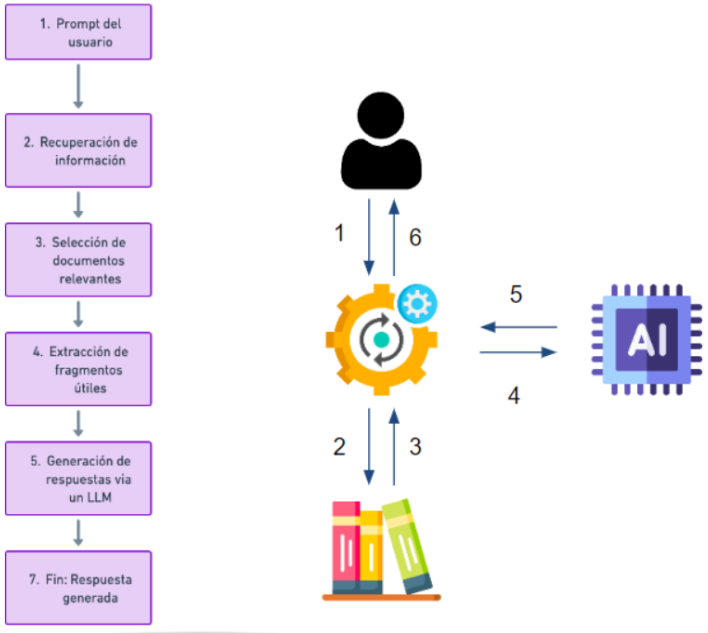

Figura 6.1: Retrieval-Augmented Generation [55].

Embeddings y Vector Stores

Un `embedding` nos permite organizar y categorizar un texto según su significado semántico. Entonces, dividimos nuestros documentos en muchos fragmentos de texto y usamos embeddings para caracterizar cada fragmento de texto según su significado semántico. Se usa un transformer de embedding para convertir un fragmento de texto en un

embedding.

Un embedding categoriza un fragmento de texto dándole una representación vectorial (coordenada). Eso significa que los vectores que están cerca unos de otros representan fragmentos de información que tienen un significado similar entre sí. Los vectores de embedding se almacenan dentro de un **vector store**, junto con los fragmentos de texto correspondientes a cada embedding.

Una vez que disponemos de un prompt, usamos el transformer de embeddings para generar un embedding (vector) a partir de él. De esta manera, podemos asociarlo con los fragmentos de texto más relevantes semánticamente, localizando otros embeddings (vectores) cercanos. De este modo, contamos con un método para emparejar nuestro prompt con otros fragmentos de texto relacionados almacenados en el vector store. En nuestro caso, utilizamos el transformador de embeddings de OpenAI, el cual utiliza el método de similitud del coseno para calcular la similitud entre documentos y una pregunta.

Con este proceso, obtenemos un subconjunto más reducido de la información relevante para nuestro prompt, lo que nos permite consultar al LLM con nuestro prompt inicial, proporcionando únicamente la información pertinente como contexto para nuestro prompt.

De esta manera, superamos la limitación de tamaño en los prompts de LLM al utilizar embeddings y un vector store para transmitir únicamente la información relevante relacionada con nuestra consulta y obtener una respuesta basada en ello.

Chunk

Antes de pasar a explicar cómo hemos desarrollado nuestro chatbot, queremos describir también cómo dividimos nuestros documentos en muchos fragmentos de texto, pues a la hora de programar el chatbot, hay que decidir qué valores le vamos a dar a algunos parámetros para dividir los documentos.

La elección de los valores de los parámetros **'chunk_size'** y **'chunk_overlap'** [56] en el procesamiento de documentos es una consideración crítica que puede impactar significativamente la calidad y la coherencia de las respuestas generadas. Como hemos explicado, estos parámetros determinan cómo se divide y se procesa el contenido del documento, lo que influye directamente en la capacidad de nuestro chatbot para comprender y responder de manera efectiva a las consultas de los usuarios.

El **'chunk_size'**, o tamaño del fragmento, define la cantidad de contenido que se procesa simultáneamente. Si optamos por un tamaño de fragmento más grande podemos capturar más contexto relevante de una vez, lo que podría mejorar la comprensión global del documento. Sin embargo, esto también puede conllevar un aumento en la carga computacional y en la memoria requerida para el procesamiento. Por otro lado, un tamaño de fragmento más pequeño ofrece un enfoque más granular, lo que resulta útil para documentos densos con detalles importantes dispersos en múltiples partes.

Por otro lado, la **'chunk_overlap'**, o superposición de fragmentos, determina cuánto contenido se comparte entre fragmentos adyacentes. Una superposición más alta puede ayudar a mantener la coherencia del contexto a lo largo de los fragmentos, lo que facilita al chatbot comprender el contenido de manera continua. Sin embargo, esto también puede generar redundancia y aumentar los recursos computacionales necesarios. Por el contrario, una superposición baja puede resultar en la pérdida de información contextual, especialmente en áreas donde el contenido es denso o complejo.

Para nuestro chatbot informativo sobre trastornos alimenticios, BoTCA, es crucial seleccionar valores que permitan capturar suficiente contexto relevante sin comprometer la eficiencia computacional. Comenzamos con un *chunk_size* moderado, en el rango de 500 a 2000. Por lo que decidimos que un valor de 1000 sería adecuado. Asimismo, es recomendable una *chunk_overlap* moderada, entre el 5 % y el 20 % del tamaño del fragmento, para garantizar una continuidad contextual adecuada. En consecuencia, le hemos dado un valor de 10.

6.1.1. Interactuando con nuestros documentos

Primeramente, la Figura 6.2 muestra cómo funciona nuestro código, para entenderlo mejor.

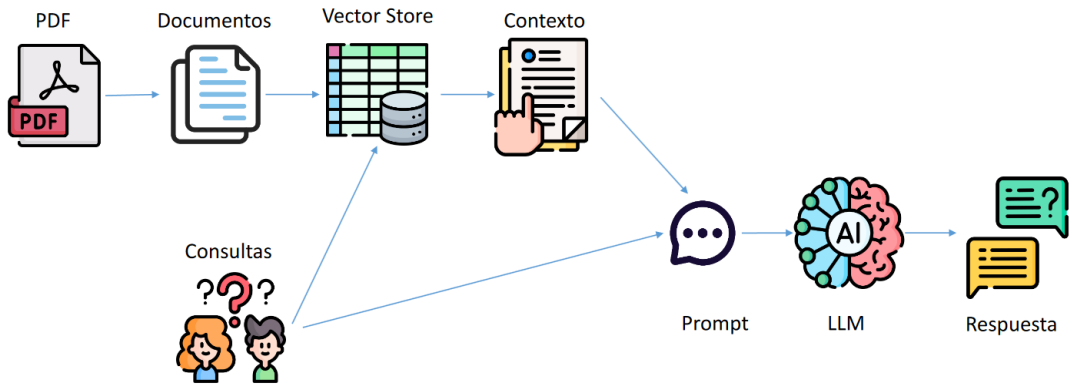

Figura 6.2: Creando un prompt usando solo la información relevante de nuestros documentos.

A continuación, explicamos los pasos seguidos para la creación de nuestro chatbot usando la API de OpenAI para el modelo GPT-3.5, LangChain y, FastAPI. Esta última la utilizamos para establecer un servidor que proporciona un formulario web donde los usuarios pueden realizar las consultas y obtener respuestas del sistema.

1. **Importaciones de módulos:** Debemos comenzar importando los módulos necesarios de FastAPI, LangChain, y otros paquetes que serán utilizados, como os, uvicorn, time, etc.

2. **Configuración de la API de OpenAI:** Se establece la clave de la API de OpenAI como una variable de entorno para autenticar las solicitudes a la API.

3. **Creación de la aplicación FastAPI:** Se crea una instancia de FastAPI para manejar las solicitudes HTTP.

4. **Carga de documentos** [1]**:** El código carga documentos desde una carpeta específica ((`.\multi-doc-chatbot\docs` en nuestro caso) utilizando diferentes cargadores de documentos, como PyPDFLoader, que carga documentos PDF, y extiende la lista documentos con los documentos cargados.

5. **División de documentos:** Los documentos cargados se dividen en fragmentos de texto más pequeños utilizando la clase CharacterTextSplitter con un tamaño de fragmento (chunk_size) y un solapamiento de fragmentos (chunk_overlap) específicos.

6. **Conversión a embedding y guardado en vector:** Los fragmentos de texto se convierten en embeddings utilizando la técnica de Chroma embeddings de LangChain y se almacenan en un vector persistente en el directorio ./data.

7. **Creación de la cadena de recuperación conversacional:** La creación de la cadena de recuperación conversacional es una parte crucial del chatbot, ya que es responsable de procesar las consultas de los usuarios y generar respuestas relevantes y coherentes:

    - **Clase ConversationalRetrievalChain:** Esta clase forma parte de la biblioteca LangChain y se utiliza para construir una cadena de procesamiento que maneja la conversación y la recuperación de información. Específicamente, en este contexto, se utiliza para implementar nuestro chatbot conversacional que puede responder a las preguntas de los usuarios basándose en la información contenida en los documentos previamente cargados.

[1] Estos documentos se pueden encontrar en el Anexo XXX

- **Modelo de lenguaje de conversación de OpenAI (ChatOpenAI):** Este componente es el encargado de generar respuestas coherentes y relevantes a partir de las preguntas de los usuarios. Utilizamos un modelo de lenguaje pre-entrenado de OpenAI, en este caso, el modelo GPT-3.5-turbo, que está diseñado para tareas de conversación.

- **Recuperador basado en embeddings de documentos:** Este componente se encarga de recuperar los fragmentos de texto relevantes de los documentos cargados que contienen información relacionada con la pregunta del usuario. Utiliza los embeddings (representaciones vectoriales) de los documentos y las consultas para calcular la similitud y recuperar los documentos más relevantes.

- **Configuraciones adicionales:** Además de estos componentes principales, la *ConversationalRetrievalChain* puede tener otras configuraciones, como especificar el número de documentos a recuperar o ajustar la temperatura en la generación de respuestas que nos permite controlar la creatividad de las respuestas generadas por el modelo de lenguaje, entre otras.

8. **Definición de rutas de la API:** Se definen las rutas para el punto de entrada principal (/) y para la consulta (/consulta/). La ruta principal genera y muestra un formulario HTML donde los usuarios pueden ingresar sus consultas. La ruta de consulta procesa las consultas enviadas desde el formulario, invoca la cadena de recuperación conversacional y devuelve la respuesta al usuario.

9. **Ejecución de la aplicación:** Si el script se ejecuta directamente, se inicia el servidor Uvicorn para manejar las solicitudes entrantes.

Al ejecutar la aplicación, nuestro sistema muestra una página principal para hacer las consultas, como podemos ver en la Figura 6.3.

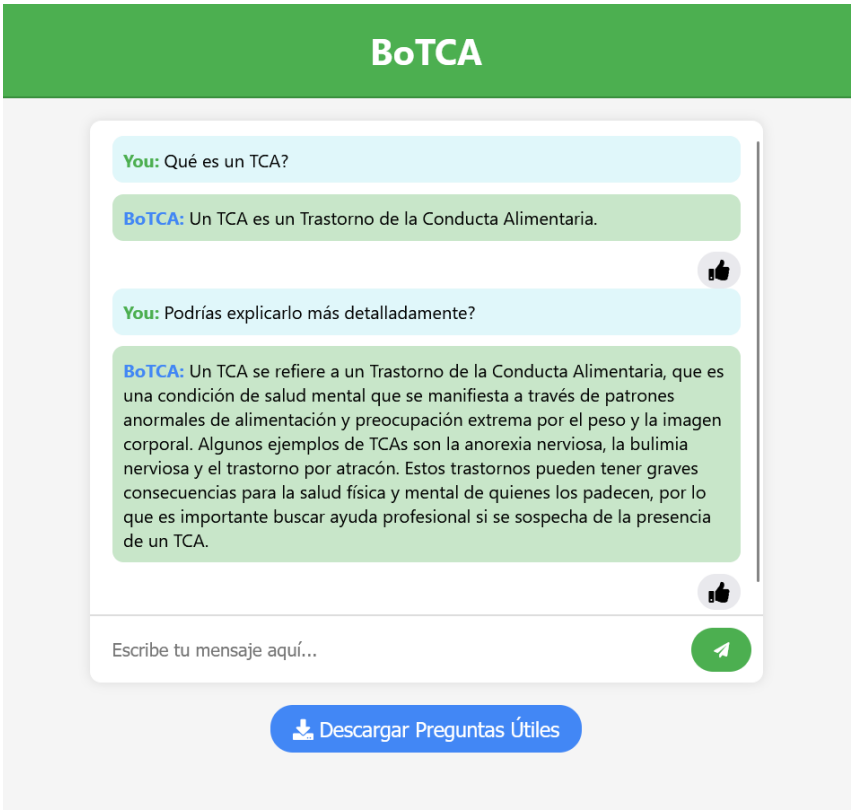

Figura 6.3: Interfaz del sistema

Si le preguntamos, la interfaz muestra las respuestas, diferenciando entre el usuario y el propio sistema. Además, el botón de enviar estará deshabilitado hasta que el usuario escriba algo en el campo de entrada. Una vez que el usuario escriba algo, el botón se habilitará y podrá enviar el mensaje. Además, nuestra interfaz tiene incorporado un botón para que, si el usuario considera útil la respuesta, se guarde esta junto con la consulta correspondiente en un archivo .csv. Puede también descargar este archivo, lo que es útil para trabajos futuros, porque se podrían utilizar estas preguntas y respuestas para ajustar el modelo, y así optimizar el sistema.

Capítulo 7

Resultados y evaluación

En este apartado, vamos a detallar los resultados obtenidos con nuestro sistema basado en RAG. Para evaluar el sistema BoTCA desarrollado, hemos generado un corpus de preguntas y respuestas (QABoTCA) que ha sido manualmente anotado por 3 evaluadores humanos. Este corpus se puede consultar en el Anexo A[1]. Aclarar que únicamente se realizará la evaluación sobre el conjunto de datos proporcionados por el modelo GPT-3.5[2]. Se ha decidido no realizar esta evaluación con los otros dos modelos utilizados, LlaMA2 y Gemma, puesto que las interacciones realizadas han proporcionado unos resultados que no alcanzan la mínima calidad como para ser considerados. En el caso de LlaMA , muchas de las respuestas generadas eran en inglés, y en el de Gemma, no era capaz de generar respuestas sintáctica y gramaticalmente correctas.

En primer lugar, debemos destacar que el análisis y la evaluación de resultados de nuestro sistema implica tres fases:

1. Carga de documentos y creación de embeddings.
2. Interacción con el usuario.
3. Análisis del texto generado.

7.1. Carga de documentos y crecaión de embeddings

En esta primera fase, hemos calculado unas estadísticas sobre el tiempo que toma la carga de documentos, división de los mismos, conversión a embeddings, y almacenamiento en vectores. Dichos datos podemos observarlos en la siguiente imagen 7.1:

[1]Se han hecho las mismas preguntas con GPT y LlaMA, aunque la evaluación es únicamente sobre GPT.

[2]Queremos resaltar que estas respuestas han sido mostradas a un psiquiatra experto en la materia que de manera global pero no exhaustiva, validó los resultados con este modelo.

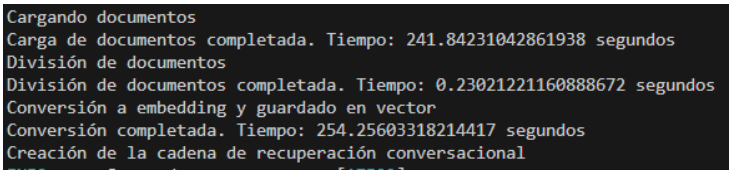

Figura 7.1: Tiempos de la primera fase

Podemos ver que la carga de documentos, la conversión a embeddings y almacenamiento en el vector nos llevará un tiempo considerable pero una vez que se haya completado esta fase, la interacción con el usuario y la generación de respuestas de nuestro sistema es inmediata.

7.2. Evaluación manual

En la segunda fase, vamos a hacer una evaluación manual de las respuestas generadas en base a las consultas del usuario. Debemos tener en cuenta que los perfiles de los tres anotadores son variados en cuanto a edad y conocimiento del tema. Además, la evaluación de cada anotador no es visible para los otros anotadores, para que no haya sesgos. Así pues, vamos a explicar la medida de evaluación escogida. Las respuestas pueden evaluarse de la siguiente manera:

- **0 - no válida/no adecuada:** respuesta incorrecta, no tiene nada que ver y es inútil. Además, puede contener contenido perjudicial y no adecuado para familias o para personas con TCA.
- **1 - no válida:** respuesta incorrecta, no tiene nada que ver y es inútil. No transmite ninguna idea perjudicial para familias o personas con TCA.
- **2 - adecuada:** respuesta que no podemos considerar inválida, pero tampoco es una respuesta del todo correcta. Se podría decir que la respuesta aunque va en la buena dirección no la ha generado a partir del contexto del sistema de recuperación de información del RAG. Esto es interesante por el nivel de generación de lenguaje de los LLMs.
- **3 - válida/no adecuada:** la respuesta es válida, generada a partir del contexto del RAG, pero contiene expresiones que pueden ser no adecuadas.
- **4 - válida/adecuada:** respuesta correcta y adecuada.

En este sentido, hemos generado 20 preguntas para que BoTCA responda y poder así evaluar estas respuestas. Podemos encontrar estas, junto con las anotaciones de cada uno de

los anotadores, en el Anexo X. Las Figuras 7.2, 7.3 y 7.4 muestran los valores asignados a las 20 preguntas por cada uno de los anotadores.

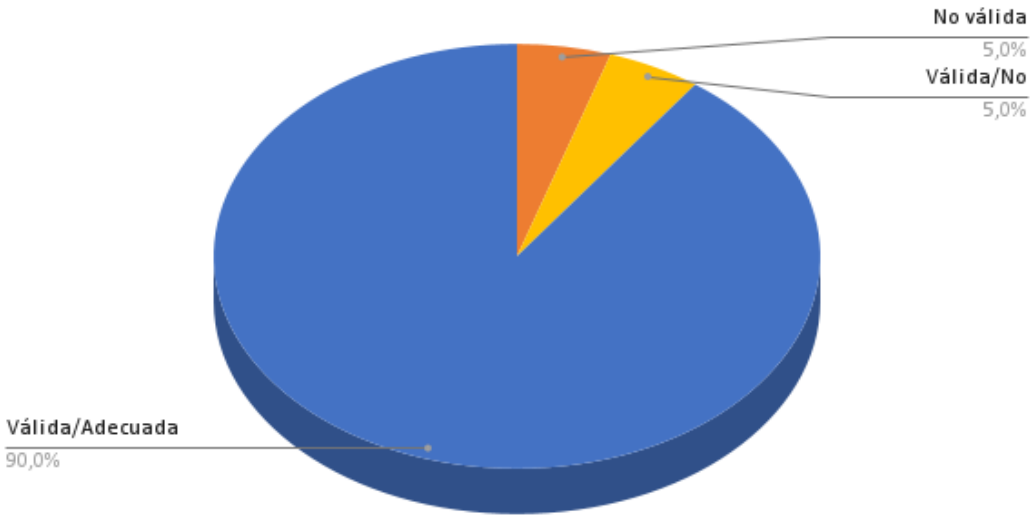

Figura 7.2: Evaluación del anotador 1

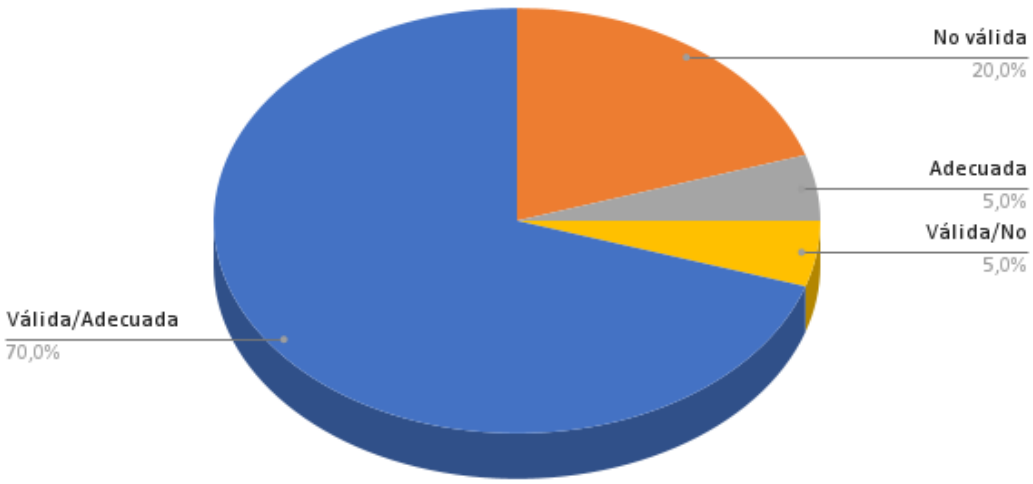

Figura 7.3: Evaluación del anotador 2

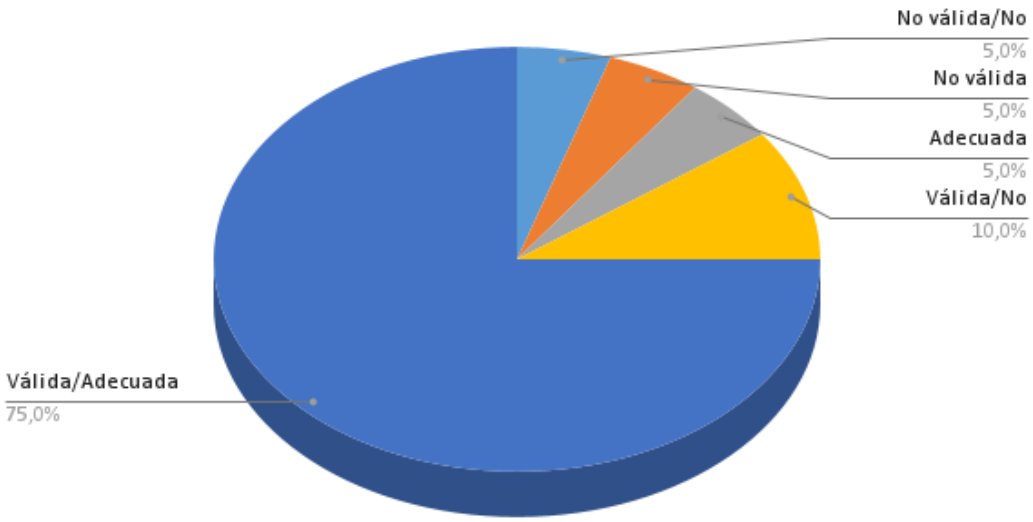

Figura 7.4: Evaluación del anotador 3

Si bien, los valores utilizados para la evaluación son categorías, también es cierto que se pueden utilizar como valores numéricos ya que 0 representa el peor valor posible y 5 el mejor valor. De esta manera, las Figuras 7.2, 7.3 y 7.4, muestran una gráfica cuantitativa de las respuestas obtenidas con BoTCA y evaluadas por los 3 anotadores.

7.3. Análisis lingüístico del corpus generado

En la última fase, vamos a hacer un análisis del texto con la herramienta Voyant [57]. Esta nos permite visualizar y analizar textos de varias maneras diferentes. En la siguientes figuras podemos observar los análisis realizados:

Nube de palabras La figura 7.5 nos proporciona una vista de aquellas palabras que se repiten con mayor frecuencia en las respuestas generadas por nuestro sistema. Nos proporciona una visión general y conveniente del contenido.

Figura 7.5: Nube de palabras

Enlaces entre términos La siguiente figura 7.6 muestra un gráfico de red de los términos con una frecuencia más alta que aparecen por proximidad. Las palabras clave se muestran en azul y las colocaciones (palabras en la proximidad) se muestran naranja.

Figura 7.6: Enlaces y colocaciones

Tendencias La Figura 7.7 muestra un gráfico lineal de las frecuencias relativas de los términos más frecuentes de las respuestas generadas por BoTCA.

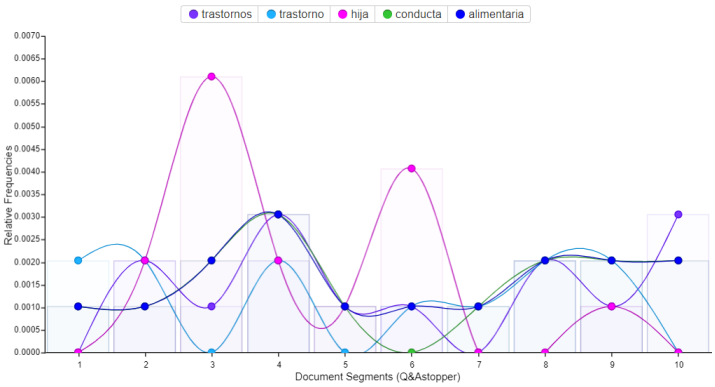

Figura 7.7: Tendencias de palabras más relevantes

Sumario En la Figura 7.8 podemos ver un sumario que nos proporciona información general del texto. Nos da información como la densidad del vocabulario, el promedio de palabras por oración, términos más frecuentes, formas de las palabras, etc.

Este corpus tiene 1 documento con 982 total de palabras y 535 formulario de palabra única. Creado hace 18 minutos atrás.

Densidad del vocabulario 0.545

Readability Index: 25.493

Promedio de palabras por oración: 33.9

Palabra más frecuente en el corpus:

- hija (16); alimentaria (16); conducta (15); trastornos (14); trastorno (12); tca (12); importante (12); comida (9); comer (9); alimentación (8); salud (7); peso (7); apoyo (7); abordar (7); ingesta (6); vida (5); sentimientos (5); proporcionado (5); profesional (5); preocupaciones (5); tanto (4); sociales (4); redes (4); problemas (4); pérdida (4); persona (4); pandemia (4); negativos (4); médico (4); mental (4); manera (4); influir (4); fundamental (4); familia (4); falta (4); buscar (4); atracón (4); atención (4); anorexia (4); alimentos (4); alimentario (4); además (4); texto (3); situación (3); significativo (3); sensación (3); relación (3); recordar (3); recomendable (3); posible (3); podría (3); personas (3); no (3); niño (3); nerviosa (3); influencia (3); impacto (3); hablar (3); fomentar (3)

Figura 7.8: Información general sobre las respuestas generadas por BoTCA

Colocaciones Esta tabla 7.9 nos ayuda a visualizar cual de los términos aparece con más frecuencia de acuerdo a su proximidad con las palabras clave a través de todo el texto.

Término	Contexto del término	Contar (contexto)
conducta	alimentaria	15
trastornos	conducta	9
alimentaria	tca	5
trastorno	conducta	4
trastorno	alimentario	4
hija	podría	2
trastornos	alimentarios	2
trastornos	alimentación	2
importante	recordar	2
hija	tome	1
hija	sufren	1

Figura 7.9: Colocaciones

Capítulo 8

Conclusiones y trabajos futuros

8.1. Conclusiones

Para finalizar este proyecto, quiero resaltar una serie de conclusiones, reflejo del desarrollo del trabajo.

Este TFG se pensó inicialmente para ayudar a las personas que sufren TCA. A lo largo del desarrollo del mismo, hemos podido comprobar que es un tema extremadamente sensible y que por lo tanto, herramientas como las que se han estudiado, analizado y trabajado, no deberían ser utilizadas de manera directa para ayudar a personas con este problema, puesto que deberían ser tratados por personal cualificado y profesional. Por ello, se derivó este trabajo hacia una solución que consideramos mucho más práctica, ajustada a la realidad, y sensible con las personas que padecen TCA, como es un sistema de información cualificado para las personas que requieren un conocimiento más profundo de esta enfermedad, sin entrar a analizar o guiar a las personas afectadas.

Por otra parte, gracias a la Inteligencia Artificial, más concretamente al Procesamiento del Lenguaje Natural, hemos podido desarrollar un chatbot informativo, denominado BoTCA, que genera respuestas a las consultas del usuario, con información confiable ya que extrae el conocimiento de una colección de textos filtrados previamente y utilizando la tecnología RAG. Como hemos podido observar, existen diferentes técnicas para mejorar el rendimiento de dicho sistema, y además, distintos modelos que no siempre generan una respuesta apropiada a esa consulta.

Gracias a esto, he podido aprender acerca de los modelos del lenguaje, la evolución de ellos, y las distintas alternativas existentes para abordar algunos de los problemas del mundo real.

Quiero destacar que he tenido la oportunidad de estudiar las últimas tendencias y desarrollos en el campo de las Tecnologías del Lenguaje. Gracias a la evolución del TFG, no solamente he implementado varios LLMs, como son GPT3.5, LLaMA2 y Gemma, sino que además he aplicado técnicas tan novedosas, como los RAG, que han permitido obtener un sistema de información completamente confiable, suprimiendo uno de los mayores problemas a los que se enfretan los LLMs, como son las alucinaciones.

Concretamente estas son las distintas habilidades y competencias que he desarrollado o mejorado durante la realización de mi TFG:

- Aplicación de los conocimientos adquiridas, en las asignaturas desarrolladas de la carrera en un proyecto práctico completo, como por ejemplo el desarrollo de interfaces, Procesamiento del Lenguaje Natural, Sistemas de Recuperación de Información, y asignaturas relacionadas con la programación.
- Estudio y aplicaicón de las técnicas más innovadoras que se están desarrollando diariamente. Quiero resaltar que incluso he incluido en mi trabajo la utilización del modelo del lenguaje Gemma, sacado al mercado tras el inicio de mi proyecto.
- Abordaje y resolución de problemas reales, aplicando distintas soluciones de manera independiente.
- Redacción de un proyecto integral, analizando, por una parte, el problema real existente, como es un TCA, y por otra, las distintas técnicas que se pueden aplicar para abordarlo, y la evaluación de resultados de dichas estas tecnologías.

Finalmente, quiero destacar que este proyecto se alinea con el Objetivo de Desarrollo Sostenible (ODS) [58] número 3: Salud y bienestar. Como explicamos en la introducción de este trabajo, este objetivo se centra en garantizar una vida saludable y promover el bienestar para todas las personas en todas las edades.

8.2. Trabajos futuros

A pesar de que se ha desarrollado un Trabajo Fin de Grado que consideramos finalizado, se han quedado abiertos algunos retos que pueden ser estudiados más detenidamente en el futuro:

- Implementación de un bot accesible de manera libre para toda la sociedad. Para ello, se requiere el apoyo y supervisión de expertos psiquiatras para su implantación en el mundo real.

- Inclusión de un sistema de recuperación de información multimodal, en el que se incluya voz e imágenes.
- Resumen de este TFG en un artículo científico que pueda ser presentado a la comunidad investigadora, ya que trata los últimos desarrollos tecnológicos basados en los LLM en genral, y los RAG en particular.
- Aplicación de las tecnologías aprendidas en otros dominios, como pueden ser la desinformación o ciberseguridad.
- Pruebas con otros modelos del lenguaje que vayan surgiendo en un futuro, y que puedan mejorar el sistema desarrollado.

Apéndice A

Corpus de preguntas y respuestas

Preguntas	gpt3.5	llama
¿qué es un TCA, y cómo se pueden manifestar?	Un Trastorno de la Conducta Alimentaria (TCA) es una enfermedad mental que se caracteriza por una preocupación excesiva por el peso, la comida, y la imagen corporal. Los TCA pueden manifestarse de diferentes maneras, siendo los más comunes la anorexia nerviosa (restricción de la ingesta de alimentos para perder peso), la bulimia nerviosa (episodios de ingesta excesiva de alimentos seguidos de métodos compensatorios como vómitos o ejercicio excesivo) y el trastorno por atracón (ingesta excesiva de alimentos sin métodos compensatorios). Otros síntomas pueden incluir obsesión por contar calorías, miedo intenso a subir de peso, distorsión de la imagen corporal, entre otros	Un TCA (Trastorno de la Conducta Alimentaria) es un patrón de comportamiento problemático y repetitivo relacionado con la alimentación, que puede afectar la salud física y mental de una persona. Los TCA pueden manifestarse de muchas maneras, como restricciones alimentarias, comportamientos compulsivos relacionados con la comida, o una preocupación excesiva por el peso y la apariencia corporal. Estos trastornos pueden tener un impacto significativo en la calidad de vida de una persona y pueden ser difíciles de tratar sin ayuda profesional. Es importante buscar ayuda de un profesional de la salud mental si se sospecha que se está experimentando un TCA.

Preguntas	gpt3.5	llama
¿qué otros trastornos son también comunes, a parte de la anorexia y bulimia?	Además de la anorexia y la bulimia, otros trastornos de la conducta alimentaria comunes incluyen el trastorno por atracón (o binge eating disorder) y el trastorno alimentario compulsivo (o compulsive eating disorder). Estos trastornos también tienen un impacto significativo en la salud física y mental de las personas que los padecen.	Además de la anorexia y la bulimia, otros trastornos alimenticios comunes incluyen el trastorno de la conducta alimentaria (TFE), el trastorno obsesivo-compulsivo (TOC) y el trastorno de la personalidad evasiva (TPE). El TFE se caracteriza por patrones de comportamiento alimentario restringidos y obsesivos, mientras que el TOC se asocia con pensamientos obsesivos y compulsivos sobre la comida y el peso. El TPE se define por una evasión emocional y una falta de conexión social. Todos estos trastornos pueden tener un impacto significativo en la salud mental y física de una persona, por lo que es importante buscar ayuda de un profesional de la salud mental si se presentan síntomas.

Preguntas	gpt3.5	llama
Tengo una hija que no quiere comer, ¿qué debería hacer?	De acuerdo con el texto proporcionado, es importante revisar si en la familia existen dificultades en la comunicación o al resolver conflictos, ya que estos problemas pueden influir en la relación de la niña con la comida. Es esencial facilitar la expresión de los sentimientos, incluidos los negativos. También se menciona que en ocasiones, los problemas con la alimentación pueden estar relacionados con dificultades emocionales no expresadas de manera directa. Por lo tanto, además de prestar atención a los comportamientos alimenticios de la hija, es importante considerar qué aspectos de su vida podrían estar resultando difíciles y no menospreciar sus preocupaciones. En el caso concreto del texto, la psicoterapeuta sugiere a la madre permitir que la hija tome la iniciativa al comer en lugar de meterle la comida en la boca.	Si tu hija no quiere comer, es importante que primero compruebas si tiene alguna enfermedad médica subyacente que pueda estar causando su falta de apetito. Una vez que hayas descartado cualquier problema médico, es importante que la ayudes a encontrar formas saludables de abordar su relación con la comida y su cuerpo. Esto puede incluir hablar con ella sobre la importancia de la nutrición y la higiene, y ayudarla a encontrar formas de actividad física que disfrute. También es importante que la apoyes emocionalmente y que le muestres que la amas y que la respetas, independientemente de su peso o su apariencia física.

Preguntas	gpt3.5	llama
¿y si no quiere salir a comerse una pizza con sus amigos?	Según el texto proporcionado, si tu hija no quiere salir a comer pizza con sus amigos, puede ser útil reflexionar sobre la relación que tiene contigo como madre. A veces, las actitudes de los hijos hacia la comida pueden estar influenciadas por sus relaciones familiares. Es importante crear un ambiente de apoyo y comprensión para que tu hija se sienta cómoda expresando sus preferencias y necesidades. ¡Difícil situación! ¿Hay algo más con lo que pueda ayudarte?	Si tu amigo no quiere salir a comer una pizza con sus amigos, es importante respetar su decisión y no presionarle para que haga algo que no desee. La presión o la coacción pueden empeorar los trastornos de la alimentación y afectar negativamente su salud mental. En su lugar, puedes ofrecerle apoyo y comprensión, y preguntarle si hay alguna razón específica por la que no quiere salir a comer. También puedes sugerirle que hable con un profesional de la salud mental para obtener ayuda y apoyo adicionales.

Preguntas	gpt3.5	llama
he notado que a veces, después de comer, está mucho tiempo en el cuarto de baño, ¿cómo puedo actuar ante eso?	Si tu hija pasa mucho tiempo en el baño después de comer y tienes preocupaciones sobre su conducta alimentaria, es importante abordar este tema con seriedad. Puedes seguir estos pasos: 1. Habla con tu hija: Comienza por hablar con tu hija de manera tranquila y comprensiva. Pregúntale cómo se siente después de comer y por qué pasa tanto tiempo en el baño. Escucha sus respuestas sin juzgar. 2. Consulta con un profesional de la salud: Considera hablar con el pediatra de tu hija o un médico de familia para obtener orientación sobre su comportamiento y posiblemente hacer una evaluación para descartar trastornos de la conducta alimentaria. 3. Busca apoyo psicológico: Si sospechas que tu hija podría tener un trastorno de la conducta alimentaria, busca la ayuda de un psicólogo o psiquiatra especializado en trastornos de la alimentación. Ellos podrán ofrecer terapia y apoyo adecuados. 4. Contacta con servicios especializados: Recuerda que es fundamental abordar este tipo de comportamientos con sensibilidad y buscar ayuda profesional para garantizar el bienestar de tu hija.	Es importante que tengas en cuenta que los trastornos de alimentación pueden tener un impacto significativo en la salud mental y física de una persona. Si notas que tu hija está pasando mucho tiempo en el cuarto de baño después de comer, es posible que esté experimentando signos de trastorno de alimentación. Es importante que le brindes un espacio seguro y respetuoso para hablar sobre sus sentimientos y preocupaciones. También puedes ayudarla a buscar la ayuda de un profesional de la salud mental para abordar los problemas subyacentes y encontrar soluciones efectivas. Recuerda que la salud mental y la alimentación están estrechamente relacionadas, y es importante que se aborden estos temas con sensibilidad y respeto.

Preguntas	gpt3.5	llama
Además, me ha dicho que se le ha retirado la regla, ¿es una consecuencia de esta enfermedad?	Sí, la pérdida de la menstruación es una consecuencia común de los trastornos de la conducta alimentaria, especialmente en casos de anorexia nerviosa. En adolescentes, la ausencia o pérdida de la menstruación durante más de tres ciclos consecutivos es un signo de alarma que puede indicar la presencia de un trastorno alimentario.	Sí, es común que las personas con trastornos de la alimentación pierdan la regla, ya que la enfermedad puede afectar la función endocrina y el sistema nervioso. Sin embargo, es importante tener en cuenta que la pérdida de la regla no es una consecuencia directa del trastorno de la alimentación, sino que puede ser un síntoma relacionado con la enfermedad. Además, la pérdida de la regla puede ser un signo de que la persona está experimentando una disminución en la función ovárica y puede ser un indicador de la gravedad de la enfermedad. Es importante que las personas con trastornos de la alimentación busquen atención médica para evaluar su salud y recibir tratamiento adecuado.

Preguntas	gpt3.5	llama
Suele estar muchas veces de mal humor, ¿cómo debo hablarle cuando me conteste mal?	Es fundamental recordar que la comunicación en la familia es clave. Puede ser útil facilitar la expresión de los sentimientos, incluidos los negativos, y tratar de entender qué aspectos en su vida pueden estar resultando difíciles. No apresurarse a quitar importancia a sus comentarios y comprender que sus preocupaciones son importantes para ella. Intenta establecer un diálogo abierto y respetuoso para abordar la situación.	It's understandable that you may feel frustrated or upset when someone responds to you in a negative or dismissive tone. However, it's important to remember that their behavior is not a reflection of your worth or the value of your question. Instead of engaging in a back-and-forth argument or trying to convince them of your point of view, it may be more helpful to take a step back and re-evaluate the situation. Here are a few tips for communicating with someone who is not responding well: 1. Stay calm and composed: It's important to keep your cool, even if the other person is being unreasonable or rude. Take a few deep breaths and try to maintain a neutral tone. 2. Avoid getting defensive: It's natural to want to defend yourself when someone is being critical or dismissive, but getting defensive will only escalate the situation. Instead, try to listen to their perspective and respond thoughtfully. 3. Use I statements: Instead of accusing the other person or making assumptions about their motivations, try using I statements to express your own feelings and needs. For example, you might say I feel frustrated when you dismiss my question without considering it rather than You're being really rude and dismissive. 4. Seek common ground: Look for areas of agreement and try to find a way forward that works for both parties. This may involve compromising or finding a middle ground. 5. Know when to walk away: If the conversation is not productive or is causing more harm than good, it may be time to walk away and seek support from someone else. Remember that you can't control how others behave, but you can control how you respond to them. By staying calm, avoiding getting defensive, using I statements, seeking common ground, and knowing when to walk away, you can communicate effectively and maintain your own well-being.

Preguntas	gpt3.5	llama
Cuando sale, bebe mucho alcohol pero no come nada ¿debo preocuparme porque llegue siempre tarde a casa?	Según el contexto proporcionado, la falta de apetito puede estar relacionada con crisis emocionales o cambios en la vida del niño. En ocasiones, los niños pueden perder temporalmente el interés por la comida debido a descubrimientos de nuevas actividades que les otorgan autonomía, como aprender a andar solos, y poner toda su atención en ello. Por lo tanto, si un niño sale, bebe mucho pero no come nada, podría deberse a diversas razones emocionales o de desarrollo. En general, es importante observar al niño y considerar su bienestar en su totalidad, no solo su alimentación. Si persisten las preocupaciones, es recomendable consultar con un profesional de la salud.	It is understandable to be concerned about the relationship between alcohol consumption and eating habits. However, it is important to note that alcohol can have a negative impact on mental health and overall well-being. It can lead to impaired judgment, increased risk-taking behavior, and can worsen symptoms of anxiety and depression. Additionally, alcohol can also affect appetite and digestion, leading to unhealthy eating habits. If you find that you are consistently drinking excessively and arriving home late, it may be helpful to consider seeking professional help to address any underlying issues that may be contributing to this behavior. A mental health professional can provide a safe and confidential space to explore these issues and develop healthier coping strategies. In terms of your eating habits, it may be helpful to focus on nourishing your body with a balanced diet that includes a variety of whole foods. It is also important to listen to your body's hunger and fullness cues, rather than relying on alcohol to suppress your appetite. By taking care of your physical and mental health, you can improve your overall well-being and reduce the risk of negative consequences associated with excessive alcohol consumption.

Preguntas	gpt3.5	llama
De vez en cuando come mucho y luego está mucho tiempo sin comer	Basándonos en las preguntas de la guía de práctica clínica sobre trastornos de la conducta alimentaria que proporcionaste, parece que tu hija podría estar experimentando comportamientos relacionados con la alimentación que podrían indicar un posible trastorno alimentario, como atracones o restricción alimentaria. Sería recomendable buscar la opinión de un profesional de la salud especializado en trastornos de la alimentación para una evaluación adecuada y un posible diagnóstico. No obstante, como asistente digital, no puedo diagnosticar a tu hija, por lo que te recomendaría consultar con un médico o un especialista en salud mental.	It sounds like you may be experiencing some difficulties with your relationship with food and your body image. It's important to understand that food is not a solution for emotional issues, and it's not healthy to use food as an anxiolytic or to compensate for feelings of frustration or insecurity. Instead, it's important to address the underlying emotional issues and develop a healthier relationship with food and your body. It's also important to note that restrictive dieting and overexercising can be harmful and can lead to a cycle of disordered eating. Instead of focusing on dieting or restricting certain foods, it may be helpful to focus on developing a balanced and nourishing relationship with food. This can involve exploring your emotional needs and finding healthy ways to meet them, such as through self-care, mindfulness, and social support. Remember that it's okay to have cravings and to indulge in foods that bring you pleasure, as long as you are doing so in a balanced and mindful way. It's also important to prioritize self-care and self-compassion, and to seek support from a mental health professional if you are struggling with disordered eating or body image issues.

Preguntas	gpt3.5	llama
Tiene otros amigos que sufren de TCA, ¿debería dejar de juntarse con ellos?	Es importante recordar que la influencia de los amigos puede ser significativa en la vida de una persona, especialmente durante la adolescencia. Si está preocupado por la influencia que los amigos de su hija que sufren de TCA pueden tener en ella, es recomendable abordar esta preocupación de manera abierta y comprensiva con su hija. Es fundamental fomentar la comunicación abierta y honesta con ella para comprender mejor la dinámica de su amistad y brindar el apoyo necesario para ayudarla a tomar decisiones saludables. En última instancia, lo importante es promover un entorno de apoyo y comprensión para su hija, independientemente de las circunstancias en las que se encuentre.	No, no deberías dejar de juntarte con tus amigos que sufren de trastornos de la alimentación. En realidad, es importante que sigas apoyándolos y brindándoles un espacio seguro para hablar sobre sus sentimientos y necesidades. Sin embargo, es importante que tengas en cuenta que cada persona con TCA tiene necesidades y expectativas diferentes, por lo que debes ser consciente de no forzar a tus amigos a hacer algo que no estén dispuestos a hacer. En lugar de ello, debes brindarles un espacio para que expresen sus necesidades y deseos, y estar listo para escuchar y apoyarlos en la medida de lo posible.

Apéndice B

Documentos sobre TCA utilizados como base de conocimiento

A continuación, se muestran los documentos que hemos seleccionado para BoTCA como base de conocimiento.

- Prevención TCA Anorexia Bulimia.
- Guía intervención TCA niños autismo TEA.
- Nutrición Trastornos Conducta Alimentaria.
- Anorexia nerviosa y nutrición.
- Guía Información Prevención TCA.
- Anorexia, bulimia y otros TCA.
- Nutrición Saludable y Prevención de TCA.
- Guía recursos tratamiento TCA (Educadores).
- Guía recursos tratamiento TCA (Jóvenes).
- Guía recursos tratamiento TCA (Padres).
- ABC. Anorexia bulimia y comedor compulsivo.
- TCA Guía de cuidados médicos.
- Alimentación Emocional.
- Alimentos Psicológicos.

- Guía Prevención TCA y sobrepeso.
- TCA Como actuar desde la familia.
- Conductas alimentarias riesgo instrumentos medición.
- Guía TCA anorexia, bulimia y obesidad.
- TCA Anorexia y bulimia (Actividades).
- Derecho de las mujeres, imagen y TCA.
- Guía sobre Trastornos de Alimentación.
- Bueno para comer.
- Aprende a comer.
- Qué hacer que la alimentación no sea un problema.
- Guía práctica clínica TCA.
- Guía de alimentación PHS.
- Guía hábitos saludable y prevención obesidad infantil.
- Guía alimentación sana dirigida a familias.
- Adolescencia y TCA, modelos televisivos.
- TCA Bulimia, anorexias, aspectos teórico-clínicos.
- TCA Guía pacientes familiares y profesionales.
- Guía alimentarias y actividad física (sobrepeso).
- Controversias sobre los TCA.
- Psicología y desordenes alimenticios.
- Mito, narrativa y trastornos alimentarios.
- Los TCA, Guía para cuidar de un ser querido.
- Herramientas de evaluación en TCA.
- Manual educación nutricional TCA para familias.

- Come o no come. Los TCA de 0 a 14 años.
- Las prisiones de la comida.
- Recomendaciones sobre los TCA.
- Conducta y TEA. Escuela de nutrición.
- Tratamiento psicológico de TCA. Manual de auto ayuda.
- Protocolo de actuación en los TCA.
- TCA, protocolo clínico.
- Proyecto prevención del trastorno por atracón.
- Psicología de la conducta alimentaria.
- Qué le pasa. Los TCA en casa.
- Material didáctico. TCA, anorexia y bulimia.
- Guía de trastornos alimenticios.
- Guía para intervención de TCA en centros de protección.
- TCA. Criterios recursos y actividades.
- Guía clínica para TCA.
- TCA. Anorexia, bulimia. Guía pacientes y familiares.
- Catálogo de preguntas y respuestas.
- Guía completa sobre los TCA en la universidad.
- Guía de TCA, respuestas sencillas a preguntas complejas.
- TCA en adolescentes: descripción y manejo.

Apéndice C

Bibliografía

[1] Organización Mundial de la Salud (OMS). Mental health: strengthening our response. https://www.who.int/es/news-room/fact-sheets/detail/mental-health-strengthening-our-response, Año de acceso.

[2] Cruz Roja Española. Salud mental, s/f.

[3] SOM Salud Mental 360. Tca: una larga evolución, 2024. Último acceso: 12 de junio de 2024.

[4] Redacción Médica. Covid: Pandemia dispara casos de anorexia y bulimia, s/f.

[5] María Jesús López Salcedo. TCA: Trastorno de la Conducta Alimentaria, January 2023. Section: blog.

[6] Xavi Pareja. Los TCA: la enfermedad que afecta a más del 70 % de los adolescentes en España, March 2023. Section: Sanidad.

[7] El Periódico de la Energía. El tca, enfermedad que afecta al 70

[8] Acab - asociación de afectados por anorexia y bulimia, s/f.

[9] Sociedad española de medicina general y de familia (semg), s/f.

[10] Chui Yi Chan and Cheuk Ying Chiu. Disordered eating behaviors and psychological health during the COVID-19 pandemic. *Psychology, Health & Medicine*, 27(1):249–256, January 2022.

[11] Fernando Fernández-Aranda, Miquel Casas, Laurence Claes, Danielle Clark Bryan, Angela Favaro, Roser Granero, Carlota Gudiol, Susana Jiménez-Murcia, Andreas Karwautz, Daniel Le Grange, Jose M. Menchón, Kate Tchanturia, and Janet Treasure.

COVID-19 and implications for eating disorders. *European Eating Disorders Review: The Journal of the Eating Disorders Association*, 28(3):239–245, May 2020.

[12] Gobierno de España. ¿qué es la inteligencia artificial (ia)? - plan de recuperación, transformación y resiliencia, s/f.

[13] Procesamiento del lenguaje natural - instituto de ingeniería del conocimiento, s/f.

[14] What is a neural network? `https://aws.amazon.com/es/what-is/neural-network/`, 2024.

[15] ¿qué es python? - amazon web services, s/f.

[16] Hugging Face. Transformers: State-of-the-art natural language processing for pytorch and tensorflow, s/f.

[17] Sebastián Ramírez. Fastapi documentation, s/f.

[18] Tutorialspoint. Fastapi with uvicorn - tutorialspoint, s/f.

[19] Autores de LangChain. Langchain: Framework for language models applications. `https://pypi.org/project/langchain/`, Año de la última versión.

[20] Autores de PyPDF. Pypdf: Python pdf library. `https://pypi.org/project/pypdf/`, Año de la última versión.

[21] Autores de docx2txt. docx2txt: Python library for extracting text from ms word .docx files. `https://pypi.org/project/docx2txt/`, Año de la última versión.

[22] OyoClass. Python-multipart documentation, s/f.

[23] OpenAI. Openai python library. `https://github.com/openai/openai-python`, Año de la última versión.

[24] Chromadb. `https://pypi.org/project/chromadb/`.

[25] Autor(es) de la biblioteca. tiktoken. `https://pypi.org/project/tiktoken/`, Año de la versión más reciente.

[26] Microsoft Corporation. Visual studio code. `https://code.visualstudio.com/`, Año de la versión más reciente.

[27] Google LLC. Google colaboratory. `https://colab.research.google.com/`, 2024.

[28] Hugging Face. Hugging face transformers documentation. https://huggingface.co/docs/transformers/index.html, Año de acceso.

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need, August 2023. arXiv:1706.03762 [cs].

[30] Codificando Bits. Bert y la transferencia del aprendizaje, Año de publicación.

[31] Amazon Web Services. Aws - what is large language model? https://aws.amazon.com/es/what-is/large-language-model/#:~:text=Los%20modelos%20de%20lenguaje%20de%20gran%20tama%C3%B1o%20(LLM)%20son%20modelos,decodificador%20con%20capacidades%20de%20autoatenci%C3%B3n., Año de acceso.

[32] Hostinger. Modelos grandes de lenguaje (llm). https://www.hostinger.es/tutoriales/modelos-grandes-de-lenguaje-llm, Año de publicación o acceso.

[33] OpenAI. Openai gpt-3 models documentation. https://platform.openai.com/docs/models, Año de acceso.

[34] Gemini de google, Año de acceso.

[35] Meta. Llama - framework for language models. https://llama.meta.com/, Año de acceso.

[36] Cohere. Cohere. https://cohere.com/, Año de acceso.

[37] Google. Google ai gemma. https://ai.google.dev/gemma?hl=es-419, Año de acceso.

[38] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *OpenAI*, 2018. alec@openai.com, karthikn@openai.com, tim@openai.com, ilyasu@openai.com.

[39] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dorottya Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan

Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Li, Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the opportunities and risks of foundation models. 2022. *These authors contributed equally to this work.

[40] Xataka. Qué es el 'qué llama' y cómo funciona: qué sabemos de la inteligencia artificial meta. *Xataka*, Febrero 2023.

[41] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Javier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodríguez, Armand Joulin, Eduardo Tumba, and Guillaume Lampe. Llama: Modelos de lenguaje básico abiertos y eficientes. *arXiv*, 2023.

[42] Meta IA. Presentación de llama: un modelo de lenguaje grande fundamental de 65 mil millones de parámetros. Meta IA, Febrero 2023. Recuperado de URL.

[43] Google. Gemma open models, 2024.

[44] S Pressman Roger. *Ingeniería de Software: Un enfoque práctico*. McGraw Hill, 2002.

[45] Wikipedia Contributors. Diagrama de comunicación — Wikipedia, La enciclopedia libre, 2024. [En línea; consultado el 13-junio-2024].

[46] ¿qué es un diagrama de despliegue uml? `https://miro.com/es/diagrama/que-es-diagrama-despliegue-uml/`. Consultado: 19 de junio de 2024.

[47] Prompting Guide. Prompting guide. `https://www.promptingguide.ai/es`.

[48] Autores no especificados. Prompting guide. `https://www.promptingguide.ai/es/techniques`, Año de acceso.

[49] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023.

[50] OpenAI. Openai fine-tuning guide. `https://platform.openai.com/docs/guides/fine-tuning`, Año de acceso.

[51] Tongji-KGLLM. Tongji-KGLLM/RAG-Survey, January 2024. original-date: 2023-12-16T10:08:34Z.

[52] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey, 2024.

[53] Hugging Face. Hugging face transformers documentation: Main classes - pipelines, 2024.

[54] John Smith. Building a multi-document reader and chatbot with langchain and chatgpt, May 15 2023.

[55] Observatorio de IA. RAG: Alucinaciones y Personalización. `https://observatorio-ia.com/rag-alucinaciones-personalizacion`, fecha.

[56] Peter Abel. What chunk size and chunk overlap should you use?, June 10 2022.

[57] Voyant Tools. `https://voyant-tools.org/`. Accessed: fecha de acceso.

[58] Departamento de Asuntos Económicos y Sociales de las Naciones Unidas. Informe sobre los objetivos de desarrollo sostenible 2023. `https://unstats.un.org/sdgs/report/2023/The-Sustainable-Development-Goals-Report-2023_Spanish.pdf`, 2023. Accedido el día de (fecha).

[59] Augusto Cortez Vásquez, Hugo Vega Huerta, Jaime Pariona Quispe, and Ana Maria Huayna. Procesamiento de lenguaje natural. *Revista de investigación de Sistemas e Informática*, 6(2):45–54, December 2009. Number: 2.

[60] Sharath Chandra Guntuku, David B Yaden, Margaret L Kern, Lyle H Ungar, and Johannes C Eichstaedt. Detecting depression and mental illness on social media: an integrative review. *Current Opinion in Behavioral Sciences*, 18:43–49, December 2017.

[61] Elizabeth M. Seabrook, Margaret L. Kern, and Nikki S. Rickard. Social Networking Sites, Depression, and Anxiety: A Systematic Review. *JMIR Mental Health*, 3(4):e5842, November 2016. Company: JMIR Mental Health Distributor: JMIR Mental Health Institution: JMIR Mental Health Label: JMIR Mental Health Publisher: JMIR Publications Inc., Toronto, Canada.

[62] Rosie Jean Marks, Alexander De Foe, and James Collett. The pursuit of wellness: Social media, body image and eating disorders. *Children and Youth Services Review*, 119:105659, December 2020.

[63] José Alberto Benítez Andrades. Clasificación automática de textos sobre Trastornos de Conducta Alimentaria (TCA) obtenidos de Twitter.

[64] Iranzu Viguria, Miguel Angel Alvarez-Mon, Maria Llavero-Valero, Angel Asunsolo del Barco, Felipe Ortuño, and Melchor Alvarez-Mon. Eating Disorder Awareness Campaigns: Thematic and Quantitative Analysis Using Twitter. *Journal of Medical Internet Research*, 22(7):e17626, July 2020. Company: Journal of Medical Internet Research Distributor: Journal of Medical Internet Research Institution: Journal of Medical Internet Research Label: Journal of Medical Internet Research Publisher: JMIR Publications Inc., Toronto, Canada.

[65] Tutorial de Google BERT para PNL con aprendizaje automático, March 2021.

[66] Building A GPT-3 Twitter Sentiment Analysis Product | Width.ai.

[67] OpenAI Platform.

[68] Guía de Ingeniería de Prompt – Nextra.

[69] Data preparation and analysis for chat model fine-tuning | OpenAI Cookbook.

[70] Data preparation and analysis for chat model fine-tuning | OpenAI Cookbook.

[71] OpenAI Platform.

[72] Preguntas frecuentes sobre trastornos alimentarios | Servicio de Salud de Castilla-La Mancha.

[73] Preguntas frecuentes sobre los Trastornos de la Conducta Alimentaria | Hospital Clínic Barcelon.

[74] TCA en SOM360 | SOM Salud Mental 360.

[75] Preguntas frecuentes de Trastornos Alimentarios.

[76] Condé Nast. Preguntas para saber si tengo un trastorno alimenticio, August 2021. Section: tags.

[77] María Ángeles Martínez Martín. Guía de trastornos de la conducta alimentaria. Respuestas sencillas a preguntas complejas.

[78] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. Retrieval-Augmented Generation for Large Language Models: A Survey, January 2024. arXiv:2312.10997 [cs].

[79] GitHub - Tongji-KGLLM/RAG-Survey.

[80] RAG | Langchain.

[81] Sami Maameri. Building a Multi-document Reader and Chatbot With LangChain and ChatGPT, January 2024.

[82] ai geek (wishesh). Building a Private AI Chatbot, November 2023.

[83] ¿Qué es un chatbot? | IBM.

[84] Madhur Prashant. DEMO: Langchain + RAG Demo on LlaMa-2–7b + Embeddings Model using Chainlit, September 2023.

[85] RAG using Llama 2, Langchain and ChromaDB.

[86] Building an LLM application - LlamaIndex 0.9.45.post1.

[87] High-Level Concepts - LlamaIndex 0.9.45.post1.

[88] How to Build a Retrieval-Augmented Generation Chatbot, November 2023.

[89] Alucinaciones en los modelos de lenguaje: ¿Qué son y cómo minimizar los riesgos?, February 2024.

[90] Alejandrina Bautista-Jacobo, Daniel González Lomelí, Daniela Guadalupe González Valencia, and Manuel Alejandro Vazquez Bautista. Trastornos de la conducta alimentaria y ansiedad en estudiantes durante la pandemia por COVID-19: Un estudio transversal. *Nutrición Clínica y Dietética Hospitalaria*, 43(2), May 2023. Number: 2 Publisher: Fundación alimentación saludable.

[91] Ana Sofía Apodaca Cabrera. Redes Sociodigitales, Mujeres Adolescentes y Trastornos de la Conducta Alimentaria: un panorama de estudio. *Calidad de Vida y Salud*, 16(1):20–39, July 2023. Number: 1.

[92] María Jesús Vargas Baldares. TRASTORNOS DE LA CONDUCTA ALIMENTARIA.

[93] Balbino Andrés Martínez Gómez. Influencia de los medios de comunicación en el desarrollo de TCA en adolescentes.

[94] Álvaro Ojeda-Martín and Griselda Herrero-Martín. Uso de redes sociales y riesgo de padecer TCA en jóvenes. 2021.

[95] Uso de las redes sociales en España: dedicamos de media 42 minutos diarios a estar en ellas.

[96] ¡Síguenos! Ángela Blanes Psicóloga en Clínica CTA Graduada en Psicología Máster en Psicología General Sanitaria Título de Experto en Trastornos de la Conducta Alimentaria Terapeuta EMDR. Los Trastornos Alimentarios y las redes sociales, May 2022.

[97] Welcome Gemma - Google's new open LLM.

[98] Bruno López Takeyas. INTRODUCCIÓN A LA INTELIGENCIA ARTIFICIAL.

[99] Alejandro Vaca. Transformers en Procesamiento del Lenguaje Natural - IIC, May 2020.

[100] Generation with LLMs.

[101] Stuart Russell. Inteligencia artificial.

[102] Juan Victoria. *CONCEPTOS Y APLICACIONES DOCENTES DE LA INTELIGENCIA ARTIFICIAL (IA) Y EL PROCESAMIENTO DEL LENGUAJE NATURAL (PLN)*. November 2022.

[103] Procesamiento de Lenguaje Natural - Hugging Face NLP Course.

[104] ¿Qué es el procesamiento del lenguaje natural (PLN)?

[105] Mireia Ribera and Oliver Díaz Montesdeoca. *ChatGPT y educación universitaria. Posibilidades y límites de ChatGPT como herramienta docente*. Editorial Octaedro, January 2024.

[106] Alan Mathison Turing. Mind. *Mind*, 59(236):433–460, 1950.

[107] Juan Salvador Victoria Mas. Conceptos y aplicaciones docentes de la inteligencia artificial (ia) y el procesamiento del lenguaje natural (pln). In *Innovación docente e investigación en Educación y Ciencias Sociales: experiencias de cambio en la metodología docente*, pages 317–326. Dykinson, 2022.

[108] Natural language processing (nlp). Sitio web, Fecha de acceso: 27 de junio de 2024. `https://www.cloudflare.com/es-es/learning/ai/natural-language-processing-nlp/`.

[109] Prompting Guide. Prompt chaining. https://www.promptingguide.ai/techniques/prompt_chaining, Año de acceso.