



Trabajo practico Final

FCEIA

PROCESAMIENTO DEL LENGUAJE
NATURAL

Febrero, 2024.

INTEGRANTES:

Segui, Eduardo.

DOCENTES:

Juan Pablo, Manson.

Alan, Geary.

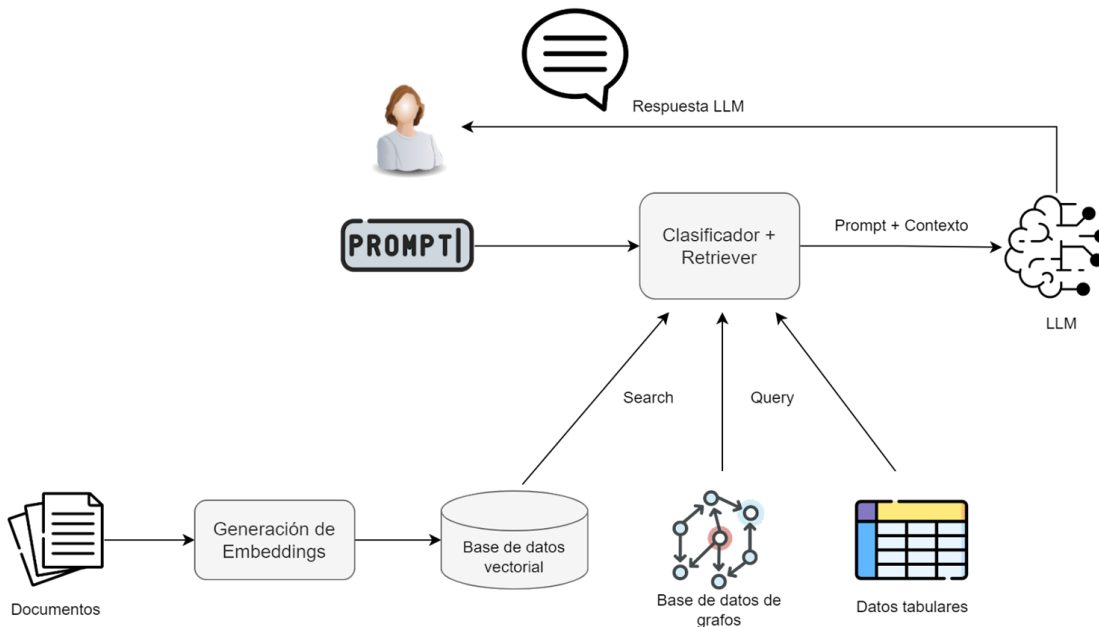
Andre Carolina, Leon Cavallo.

Ariel, D` Alessandro.

Ejercicio 1 – RAG - CHATBOT

Crear un chatbot experto en un tema a elección, usando la técnica RAG (Retrieval Augmented Generation). Como fuentes de conocimiento se utilizarán al menos las siguientes fuentes:

- ⑩ Documentos de texto
- ⑩ Datos numéricos en formato tabular (por ej., Dataframes, CSV, sqlite, etc.)
- ⑩ Base de datos de grafos (Online o local)



El sistema debe poder llevar a cabo una conversación en lenguaje español. El usuario podrá hacer preguntas, que el chatbot intentará responder a partir de datos de algunas de sus fuentes. El asistente debe poder clasificar las preguntas, para saber qué fuentes de datos utilizar como contexto para generar una respuesta.

DATOS TABULARES:

Aca voy a usar un csv que contiene la informacion de precios en dolares de departamentos por barrio de la ciudad de buenos aires. La idea es transformar el dataframe en string para poder alimentar el moldelo con esta informacion.

DOCUMENTOS DE TEXTO:

Voy a usar tres pdf que fueron descargados de la pagina de la ciudad de buenos aires: <https://buenosaires.gob.ar/cultura/patrimonio-de-la-ciudad/publicaciones/libros>.

Los libros cuentan la historia y el contexto historico de tres barrios de la ciudad de buenos aires: CHACARITA, RETIRO, COGHLAN.

A los documentos de texto los inscrusto en una base de datos chromadb pero antes se le

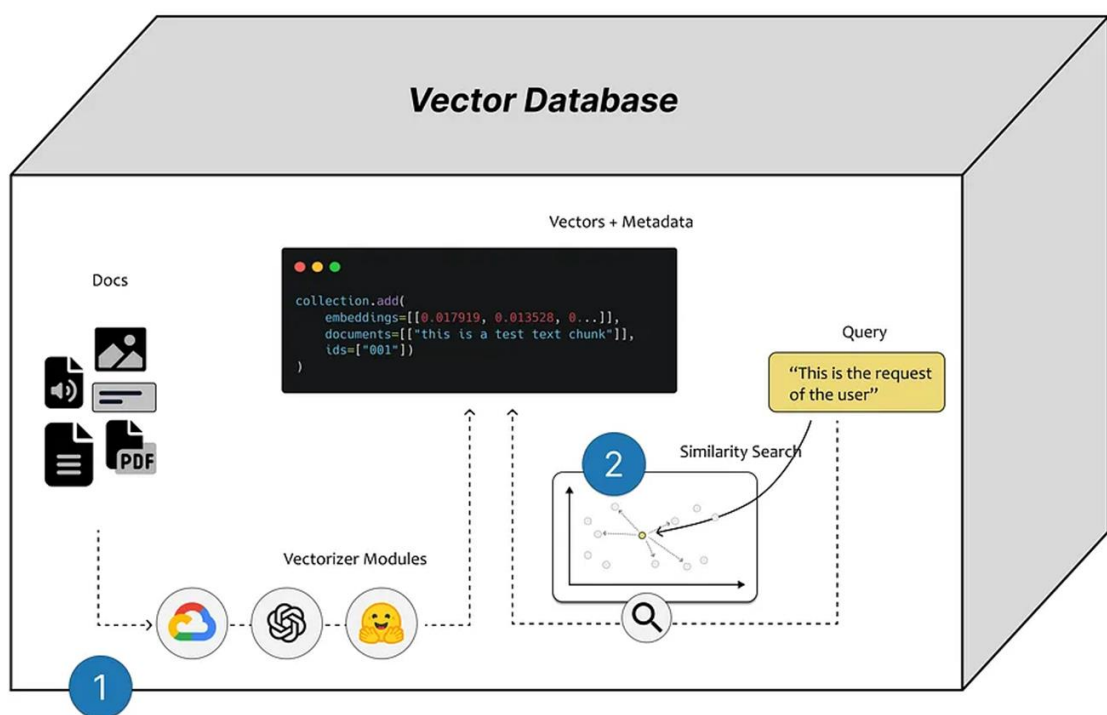
hace un pre-procesamiento de limpieza y se los divide en fragmentos para mejorar los resultados esperados por nuestro modelo, le realizo los embedding para capturar el significado semantico del texto.

ChromaDB es una base de datos vectorial de código abierto diseñada para almacenar embeddings vectoriales y desarrollar y construir aplicaciones

de modelos de lenguaje de gran tamaño. ChromaDB nos

ofrece una herramienta para realizar las siguientes funciones:

1. Almacenar embeddings y sus metadatos con identificadores.
2. Incrustar documentos y consultas.
3. Buscar embeddings.



1 Text/Images/Audio to Vectors
using Embeddings Models

2 Speed up similarity search
using Approximate Nearest
Neighbor Techniques

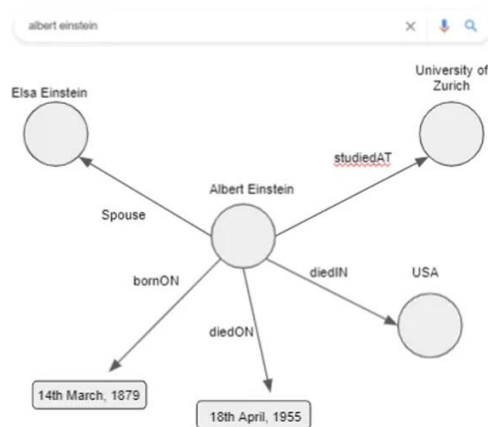
BASE DE DATOS DE GRAFOS:

Voy a consultar de Bdpedia que es base de datos de grafo de conocimiento coidgo abierto.

Las principales ventajas de un grafo de conocimiento son:

- * Datos heterogéneos: admite diferentes tipos de entidades (persona, fecha, pintura, etc.) y relaciones (gusta, nacido en, etc.).
- * Modelar información del mundo real: más cercano al modelo mental de nuestro cerebro del mundo (representa información como lo hace un humano normal).
- * Realizar razonamiento lógico: recorrer los grafos en un camino para hacer conexiones lógicas (el padre de A es B y el padre de B es C, por lo tanto, C es el abuelo de A).

Google Knowledge Panel



Albert Einstein
Theoretical physicist

Albert Einstein was a German-born theoretical physicist, widely acknowledged to be one of the greatest physicists of all time. Einstein is known widely for developing the theory of relativity, but he also made important contributions to the development of the theory of quantum mechanics. [Wikipedia](#)

Born: 14 March 1879, Ulm, Germany
Died: 18 April 1955, Penn Medicine Princeton Medical Center, New Jersey, United States
Spouse: [Eisa Einstein](#) (m. 1919–1936), [Mileva Marić](#) (m. 1903–1919)
Education: [University of Zurich](#) (1905), [ETH Zürich](#) (1896–1900), [MORE](#)

Books View 35+ more

- [Relativity : the special a...](#) 1916
- [The World As I see it](#) 1934
- [Out of My Later Years](#) 1950
- [The Evolution of Physics](#) 1938

Quotes View 7+ more

Imagination is more important than knowledge.

If you can't explain it simply, you don't understand it well enough.

Life is like riding a bicycle. To keep your balance you must keep moving.

People also search for View 15+ more

- [Eduard Einstein](#)
- [Isaac Newton](#)
- [Eisa Einstein](#)
- [Stephen Hawking](#)

SABER LA FUENTE:

Utilizo ejemplos de pocos disparos para determinar la fuente de datos más probable para responder la consulta del usuario. Esto me es útil en el RAG para que el modelo sepa cuál es la fuente más relevante y recupere la información que necesitamos para responderle al usuario.

RAG:

La Generación Aumentada por Recuperación (RAG) es el concepto de proporcionar a los Modelos de Lenguaje de Gran Escala (LLMs) información adicional proveniente de

una fuente de conocimiento externa. Esto les permite generar respuestas más precisas y contextuales, reduciendo las alucinaciones.

Los LLMs de última generación son entrenados con grandes cantidades de datos para lograr un amplio espectro de conocimiento general almacenado en los pesos

de la red neuronal (memoria paramétrica). Sin embargo, al solicitar a un LLM

que genere una respuesta que requiere conocimientos que no estaban incluidos en sus datos de entrenamiento, como información más reciente, propietaria o específica

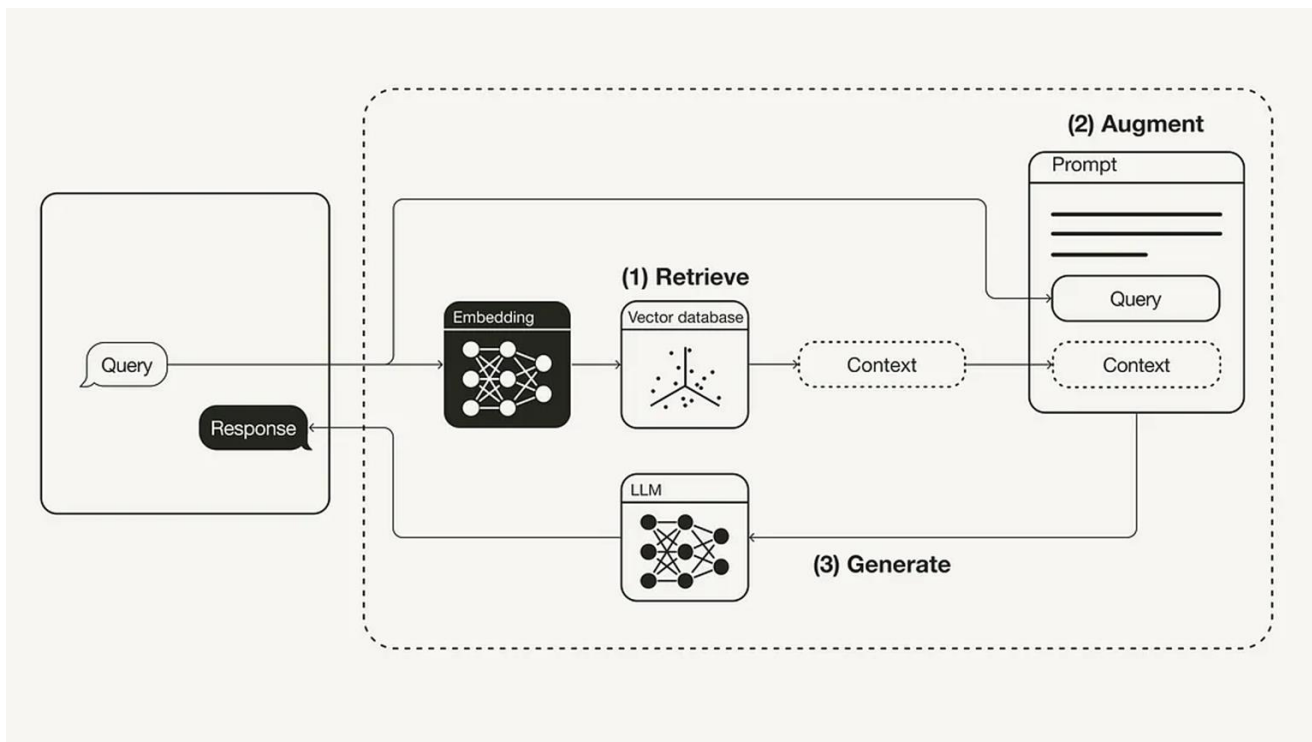
de un dominio, puede llevar a inexactitudes fácticas (alucinaciones), o bien

puede no conocer la respuesta.

Por lo tanto, es importante cerrar la brecha entre el conocimiento general del LLM y cualquier contexto adicional para ayudar al LLM a generar completaciones más precisas y contextualizadas, al tiempo que se reducen las alucinaciones. Tradicionalmente, las redes neuronales se adaptan a información específica de dominio o propietaria mediante el ajuste fino del modelo. Aunque esta técnica es efectiva, también es intensiva en cómputo, costosa y requiere experiencia técnica, lo que la hace menos ágil para adaptarse a la información en constante evolución.

Veamos las tres etapas del RAG:

- ⑩ **Recuperar:** La consulta del usuario se utiliza para obtener contexto relevante de una fuente de conocimiento externa. Para ello, la consulta del usuario se incrusta con un modelo de embeddings en el mismo espacio vectorial que el contexto adicional en la base de datos vectorial. Esto permite realizar una búsqueda de similitud y se devuelven los k objetos de datos más cercanos de la base de datos vectorial.
- ⑩ **Aumentar:** La consulta del usuario y el contexto adicional recuperado se introducen en una plantilla de solicitud.
- ⑩ **Generar:** Finalmente, la solicitud aumentada con la recuperación se alimenta al LLM.



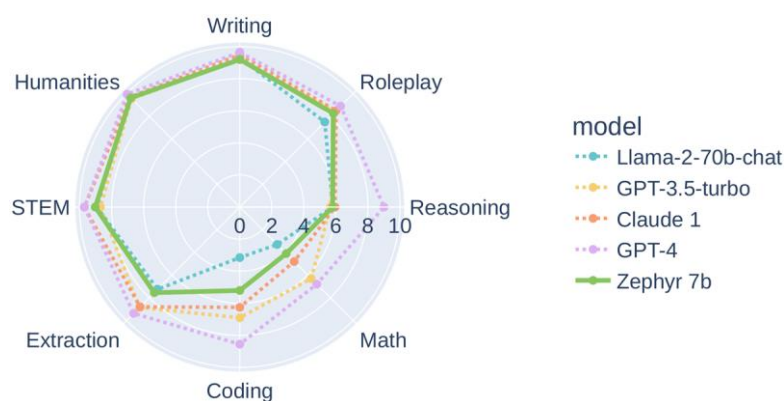
Chatbots basados en IA:

Los chatbots basados en inteligencia artificial (IA), particularmente los que emplean Modelos de Lenguaje a Gran Escala (LLMs), son sistemas avanzados de procesamiento de lenguaje natural que se fundamentan en redes neuronales profundas.

El modelo que vamos a usar es Zephyr-7B Beta.

Descripcion del modelo

- ⑩ **Tipo de modelo:** un modelo similar a GPT de parámetros 7B ajustado en una combinación de conjuntos de datos sintéticos disponibles públicamente.
- ⑩ **Idioma(s) (PNL):** Principalmente inglés(multi-lingue)
- ⑩ **Licencia:** MIT
- ⑩ **Ajustado desde el modelo:** mistralai/Mistral-7B-v0.1



Si bien el modelo es multilingüe, fue importante darle la instrucción de que responda en español porque a veces lo hacía en inglés.

```

0s [53] # Esta función prepara el prompt en estilo QA
def prepare_prompt(query_str: str, context_info: list):
    TEXT_QA_PROMPT_TMPL = (
        "La información de contexto es la siguiente:\n"
        "-----\n"
        "{context_str}\n"
        "-----\n"
        "Dada la información de contexto anterior, y sin utilizar conocimiento previo, responde solamente con la información de contexto.\n"
        "Pregunta: {query_str}\n"
        "Respuesta: "
    )
    # Construimos el contexto de la pregunta
    context_str = context_info
    messages = [
        {
            "role": "system",
            "content": "Eres un asistente útil que siempre responde con respuestas veraces, útiles y basadas en hechos.",
        },
        {"role": "user", "content": TEXT_QA_PROMPT_TMPL.format(context_str=context_str, query_str=query_str)},
    ]
    final_prompt = zephyr_instruct_template(messages)

```

Al llamar al modelo fue importante ir modificando los parametros como la temperatura, que lo uso en mes de 0,5, tambien se ajusto el top-k y top-p

para que el modelo no deleire.

CONCLUSION:

Para desarrollar un chatbot utilizando inteligencia artificial, es crucial seleccionar bases de datos que alimenten al modelo con información contextual. Además, la información utilizada para crear estas bases de datos debe ser procesada correctamente. Para afinar nuestro chatbot, es necesario implementar un RAG. Al invocar los LLMs, debemos establecer ciertos parámetros y brindarle instrucciones específicas al modelo para que responda preferiblemente con información contextual.

Ejercicio 2 - Agentes

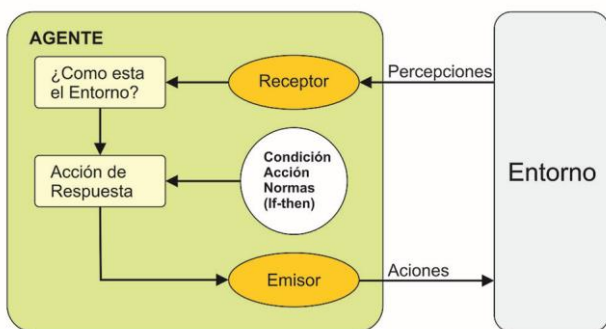
Realice una investigación respecto al estado del arte de las aplicaciones actuales de agentes inteligentes usando modelos LLM libres.

Plantee una problemática a solucionar con un sistema multiagente. Defina cada uno de los agentes involucrados en la tarea. Es importante destacar con ejemplos de conversación, la interacción entre los agentes.

Realice un informe con los resultados de la investigación y con el esquema del sistema multiagente, no olvide incluir fuentes de información.

AGENTES INTELIGENTES:

Un agente inteligente es una entidad que actúa de manera inteligente; percibe su entorno, toma acciones de manera autónoma para lograr objetivos y puede mejorar su desempeño con el aprendizaje o la adquisición de conocimientos.



Los agentes inteligentes representan la única tecnología que permite la personalización, la adaptabilidad o el autoaprendizaje de las máquinas y son ya una realidad cotidiana para la población.

Utilizar el asistente de voz de nuestro teléfono móvil, pedirle a Alexa que ponga una canción, atender las sugerencias de producto que nos hacen páginas como Amazon u obtener información a través de Google Maps son tan solo algunos ejemplos de cómo los **agentes inteligentes** forman ya parte de la vida diaria de millones de personas en todo el mundo.

La adaptación basada en la experiencia, el análisis de los índices de errores o éxitos, la resolución de problemas en tiempo real y la utilización de almacenamiento y recuperación basados en la memoria son las características comunes de los agentes inteligentes. Se trata, por lo tanto, de **programas capaces de tomar decisiones** o de realizar un determinado servicio en función de su entorno y sus expectativas.

Son entes con cierto nivel de autonomía, lo que les permite ejecutar labores específicas, predecibles y repetitivas. Además, **poseen la capacidad de aprender**, de ahí su denominación de inteligentes. Las dos funciones principales de los agentes inteligentes incluyen la percepción y la acción. La percepción se lleva a cabo a través de sensores, mientras que las acciones se inician por medio de los actuadores.

AGENTES INTELIGENTES CON LLM:

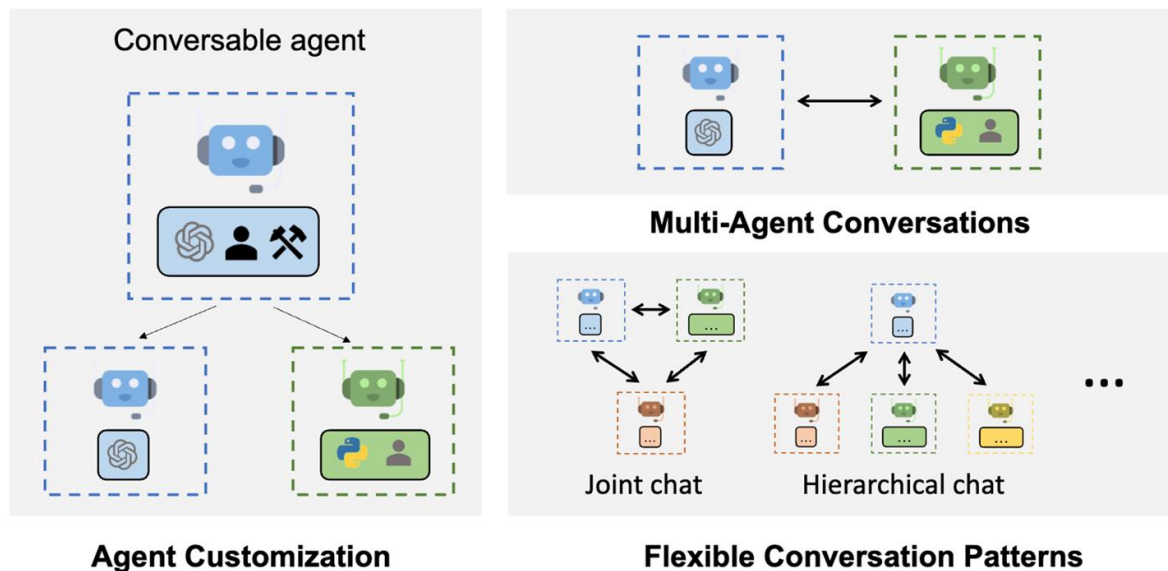
En el panorama en constante evolución de la inteligencia artificial, los modelos de lenguajes grandes (LLM, por sus siglas en inglés), como la serie GPT de OpenAI, han surgido como fuerzas transformadoras, dando forma a industrias que van desde la atención al cliente hasta la creación de contenido. Estos modelos, impulsados por extensos conjuntos de datos y algoritmos

sofisticados, tienen la notable capacidad de generar texto similar a un humano basándose en indicaciones proporcionadas. Sin embargo, su potencial se extiende mucho más allá de los productos individuales.

La llegada de los LLM, ha marcado el comienzo de una nueva era de innovación. Estos modelos han redefinido la interacción persona-computadora imitando patrones del lenguaje humano y generando respuestas coherentes y contextualmente relevantes. Sin embargo, la verdadera frontera de la IA reside en la resolución colaborativa de problemas, lo que requiere alejarse de los límites de los resultados singulares de un LLM. Aquí es donde entran en juego los LLM con múltiples agentes, que ofrecen un enfoque novedoso para abordar desafíos complejos.

LLM de agentes múltiples:

El LLM multiagente, en esencia, implica la fusión de múltiples agentes impulsados por LLM que trabajan en conjunto. A diferencia de la entidad singular tradicional que proporciona respuestas, los sistemas multiagente constan de varios agentes de IA, cada uno de los cuales se especializa en distintos dominios, lo que contribuye a la resolución integral de problemas. Esta sinergia colaborativa da como resultado soluciones más matizadas y efectivas.



Ventajas de los LLM de agentes múltiples:

Experiencia mejorada

Una de las principales ventajas de emplear LLM con múltiples agentes es la capacidad de aprovechar una amplia gama de conocimientos. Cada agente del sistema posee conocimiento especializado en su respectivo campo, lo que le permite brindar respuestas profundas y precisas. Esta amplitud de experiencia garantiza que las soluciones generadas sean integrales y estén bien informadas.

Resolución de problemas mejorada

Los desafíos complejos a menudo requieren un enfoque multifacético. Los LLM con múltiples agentes se destacan en este sentido al combinar la inteligencia colectiva de sus agentes

constituyentes. Al aprovechar sus fortalezas individuales, estos sistemas pueden analizar múltiples problemas y ofrecer soluciones innovadoras que un solo LLM tendría dificultades para igualar.

Robustez y Fiabilidad

La redundancia y la confiabilidad son factores críticos en las soluciones impulsadas por IA. Los LLM de múltiples agentes mitigan el riesgo de fallas en un solo punto. Si un agente encuentra un problema o limitación, otros pueden intervenir, garantizando una funcionalidad continua y reduciendo la probabilidad de errores o imprecisiones.

Adaptabilidad

En un mundo dinámico, la adaptabilidad es primordial. Los LLM con múltiples agentes pueden evolucionar con el tiempo, con nuevos agentes integrados perfectamente para abordar los desafíos emergentes. Esta flexibilidad los hace muy adecuados para aplicaciones en diversas industrias, desde la atención médica hasta las finanzas.

Aplicaciones de LLM de agentes múltiples:

Cuidado de la salud

En el campo médico, la información precisa y oportuna es vital. Los LLM de agentes múltiples pueden brindar experiencia bajo demanda en áreas que van desde el diagnóstico hasta las opciones de tratamiento, mejorando la atención y los resultados del paciente.

Finanzas

Las instituciones financieras pueden aprovechar el poder de los LLM de múltiples agentes para analizar las tendencias del mercado, evaluar estrategias de inversión y ofrecer asesoramiento financiero personalizado, mejorando sus servicios y la satisfacción del cliente.

Educación

Los LLM con múltiples agentes pueden revolucionar la educación al brindar a los estudiantes acceso a un conjunto diverso de expertos en la materia, ofrecer experiencias de aprendizaje personalizadas y fomentar el crecimiento académico.

Atención al cliente

Al mejorar los servicios de atención al cliente, los LLM de múltiples agentes pueden abordar consultas, resolver problemas y proporcionar información valiosa rápidamente, mejorando la experiencia general del cliente.

Superando desafíos

Si bien los LLM con múltiples agentes son inmensamente prometedores, no están exentos de desafíos. Las complejidades de la integración, el intercambio de datos y la garantía de una coordinación perfecta entre los agentes son algunos de los obstáculos que deben superarse

para desbloquear todo su potencial.

El futuro de los LLM de agentes múltiples

De cara al futuro, el futuro de los modelos de lenguajes grandes multiagente parece prometedor. Es probable que los avances en curso en la investigación y la tecnología de la IA mejoren aún más las capacidades de estos sistemas colaborativos. Aquí hay algunas posibilidades interesantes en el horizonte:

Experiencia en dominios cruzados

Los LLM de agentes múltiples podrían evolucionar para poseer experiencia en dominios cruzados, lo que les permitiría integrar perfectamente conocimientos de diversos campos. Esto les permitiría abordar con eficacia desafíos aún más complejos e interdisciplinarios.

Colaboración en tiempo real

La capacidad de los LLM de múltiples agentes para colaborar en tiempo real podría conducir a una resolución de problemas instantánea y sobre la marcha. Ya sea gestión de crisis, respuesta de emergencia o toma rápida de decisiones, estos sistemas podrían desempeñar un papel fundamental en situaciones críticas.

Consideraciones éticas

A medida que los LLM con múltiples agentes se vuelven más prominentes en los procesos de toma de decisiones, las consideraciones éticas pasarán a primer plano. Garantizar la equidad, la rendición de cuentas y la transparencia en sus acciones será imperativo para mantener la confianza y los estándares éticos.

Implementación de LLM de agentes múltiples

Ahora que entendemos el potencial y las ventajas de los modelos de lenguaje grande multiagente, el siguiente paso es explorar cómo las organizaciones pueden implementar estos sistemas de manera efectiva.

Integración de datos

La integración de datos de diversas fuentes es crucial para que los LLM con múltiples agentes funcionen de manera óptima. Las organizaciones deben desarrollar canales de datos sólidos que faciliten el intercambio fluido de información entre los agentes. Esto garantiza que cada agente tenga acceso a los datos más actualizados y relevantes.

Mecanismos de coordinación

Establecer mecanismos eficientes de coordinación entre los agentes es fundamental. Esto implica definir reglas, protocolos y canales de comunicación para garantizar que los agentes trabajen de manera coherente hacia objetivos comunes. La coordinación eficaz mejora el rendimiento general del sistema multiagente.

Monitoreo y Mantenimiento

El monitoreo y el mantenimiento continuos son vitales para el éxito a largo plazo de los LLM de múltiples agentes. Las organizaciones deben implementar herramientas de monitoreo para rastrear el desempeño de los agentes individuales y del sistema en su conjunto. Se necesitan actualizaciones y mantenimiento periódicos para abordar problemas, mejorar la eficiencia y adaptarse a los requisitos cambiantes.

Consideraciones éticas en LLM de agentes múltiples

A medida que crece la adopción de LLM con múltiples agentes, las consideraciones éticas se vuelven cada vez más importantes. Las organizaciones deben ser proactivas al abordar estos desafíos éticos:

Sesgo y equidad

Es primordial garantizar que los agentes dentro de un sistema de múltiples agentes brinden respuestas justas e imparciales. Las organizaciones deben implementar medidas para detectar y mitigar sesgos en los datos y algoritmos utilizados por los agentes.

Transparencia

Mantener la transparencia en los procesos de toma de decisiones de los LLM de múltiples agentes es esencial para generar confianza. Las organizaciones deberían poder explicar cómo y por qué el sistema genera respuestas o recomendaciones específicas.

Responsabilidad

Deben establecerse líneas claras de rendición de cuentas dentro de los sistemas de múltiples agentes. Las organizaciones deben definir quién es responsable de las acciones y decisiones tomadas por los agentes, especialmente en los casos en que estas decisiones tengan consecuencias significativas.

El camino por delante

El viaje al mundo de los modelos de lenguajes grandes multiagente apenas comienza. A medida que estos sistemas se vuelvan más frecuentes y sofisticados, continuarán remodelando industrias y redefiniendo lo que es posible en el ámbito de las soluciones impulsadas por la IA.

El impacto global de los LLM con múltiples agentes

La influencia de los modelos de lenguaje grande multiagente se extiende más allá de las organizaciones e industrias individuales. Su impacto global es significativo y de gran alcance:

Comunicación transcultural

Los LLM de agentes múltiples rompen las barreras del idioma y facilitan la comunicación intercultural. Pueden proporcionar servicios de traducción e interpretación en tiempo real, lo que facilita la interacción y colaboración de personas de diversos orígenes lingüísticos.

Resolución de problemas globales

En un mundo interconectado que enfrenta desafíos globales complejos, los LLM de múltiples agentes pueden desempeñar un papel crucial en la resolución de problemas. Pueden analizar vastos conjuntos de datos relacionados con cuestiones como el cambio climático, la salud pública y las disparidades socioeconómicas, ofreciendo conocimientos y soluciones que trascienden fronteras.

Acceso a la experiencia

Las áreas remotas y desatendidas pueden beneficiarse de la experiencia de los LLM de múltiples agentes. Estos sistemas pueden proporcionar acceso virtual a expertos médicos, educativos y profesionales, cerrando brechas en conocimientos y servicios en regiones remotas.

Desafíos en el escenario global

Si bien los LLM con múltiples agentes son inmensamente prometedores, también presentan desafíos a nivel global:

Seguridad y privacidad

La adopción global de estos sistemas genera preocupaciones sobre la seguridad y privacidad de los datos. Garantizar que la información confidencial siga protegida y que se respete la privacidad del usuario es un desafío constante que requiere cooperación internacional.

Estandarización

Los estándares y regulaciones globales para los LLM con múltiples agentes aún están evolucionando. Armonizar estos estándares en diferentes países y regiones es esencial para garantizar un uso responsable y ético de la tecnología.

Impacto socioeconómico

El uso generalizado de LLM con múltiples agentes puede alterar los mercados e industrias laborales tradicionales. Abordar el impacto socioeconómico de estos cambios, incluido el posible desplazamiento de empleos y la reconversión laboral, es una cuestión global compleja.

Colaboración para un futuro mejor

Para maximizar los beneficios globales de los modelos de lenguajes grandes multiagente y al mismo tiempo mitigar los desafíos, la colaboración internacional es esencial. Los gobiernos, organizaciones e investigadores deben trabajar juntos para:

- ⑩ Establecer pautas y estándares éticos para el desarrollo y uso de LLM multiagente.
- ⑩ Invertir en medidas de ciberseguridad para salvaguardar los datos y la privacidad en un contexto global.
- ⑩ Fomentar la investigación y la innovación para abordar el impacto socioeconómico y las oportunidades creadas por estos sistemas.
- ⑩ Garantizar que los beneficios de los LLM con múltiples agentes lleguen a las comunidades desatendidas en todo el mundo.

Dónde encontrar Grandes Modelos de Lenguaje:

En junio de 2020, OpenAI lanzó GPT-3 como servicio, impulsado por un modelo de 175 mil millones de parámetros que puede generar texto y código con breves indicaciones escritas.

En 2021, NVIDIA y Microsoft desarrollaron Megatron-Turing Natural Language Generation 530B, uno de los modelos más grandes del mundo para comprensión de lectura e inferencia de lenguaje natural, que facilita tareas como resumir y generar contenido.

Y HuggingFace presentó el año pasado BLOOM, un modelo de lenguaje grande y abierto que puede generar texto en 46 lenguajes naturales y más de una docena de lenguajes de programación.

Otro LLM, Codex, convierte texto en código para ingenieros de software y otros desarrolladores.

NVIDIA ofrece herramientas para facilitar la creación y la implementación de grandes modelos de lenguaje:

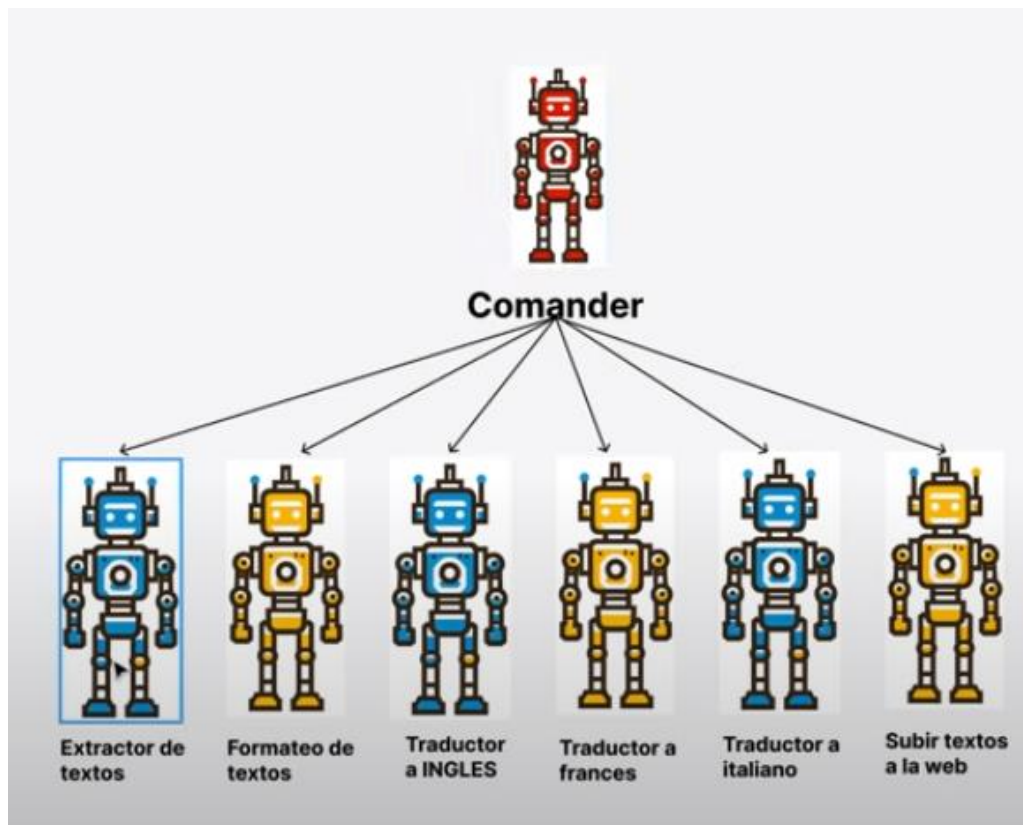
⑩ El servicio NVIDIA NeMo LLM proporciona una ruta rápida para personalizar modelos de lenguajes grandes e implementarlos a escala utilizando la API de nube administrada de NVIDIA o a través de nubes públicas y privadas.

⑩ NVIDIA NeMo Megatron, parte de la plataforma NVIDIA AI, es un marco para el entrenamiento y la implementación fácil, eficiente y rentable de grandes modelos de lenguaje. Diseñado para el desarrollo de aplicaciones empresariales, NeMo Megatron proporciona un workflow integral para el procesamiento automatizado de datos distribuidos; entrenar tipos de modelos personalizados a gran escala, incluidos GPT-3 y T5; y el despliegue de estos modelos para la inferencia a escala.

⑩ NVIDIA BioNeMo es un servicio administrado específico de dominio y un marco para grandes modelos de lenguaje en proteómica, moléculas pequeñas, ADN y ARN. Se basa en NVIDIA NeMo Megatron para entrenar e implementar grandes modelos de IA de transformadores biomoleculares a escala de supercomputación.

DESAFIO A RESOLVER CON MULTI AGENTES:

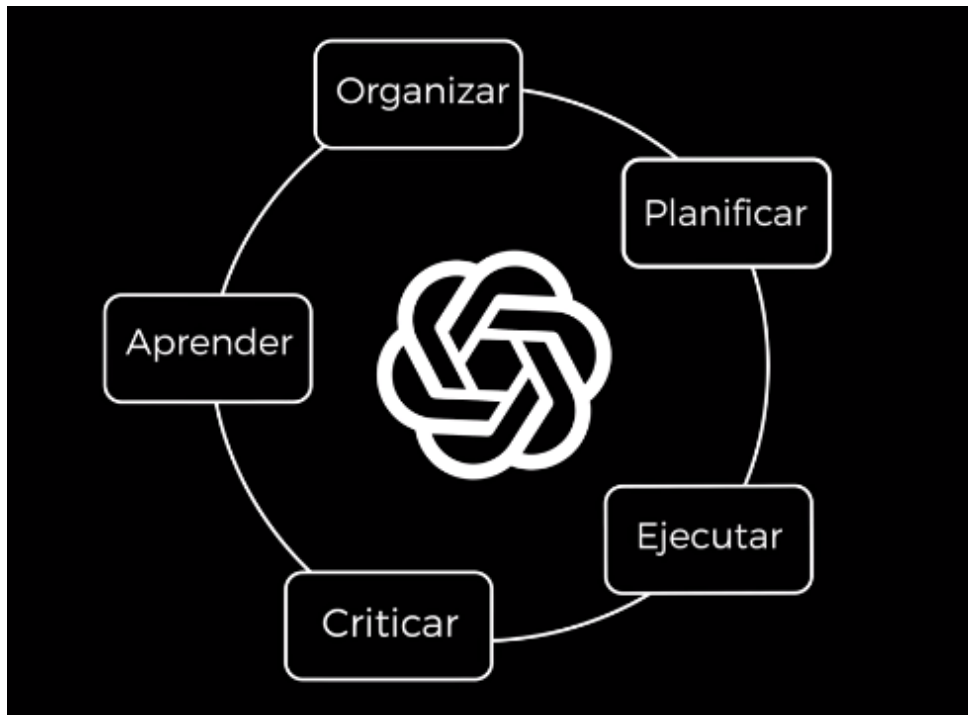
Supongamos que soy una empresa de aquí de Rosario que fabrica alfajores de muy buena calidad y quiero expandir mi negocio de forma rápida a varios países del mundo. Hoy en día, la forma más rápida sería tener una página web para llegar a las principales ciudades de América. Debería traducir mi página a varios idiomas para poder vender en otros países. Todo este nuevo desafío me podría llevar mucho tiempo, bueno aquí es donde usaría los multiagentes para resolver mi problema.



La traducción de la página a varios idiomas se puede hacer teniendo varios agentes: un comandante, extractor de textos, formateador de textos, traductor de inglés, traductor de francés, traductor de italiano y subidor de textos a la web. Entonces, el comandante hablará con el extractor de texto, el cual sacará los textos de mi página. Luego, el formateador de textos formateará los textos y se los pasará al comandante, quien los enviará a cada traductor. Después, el comandante enviará los textos traducidos para subirlos a la web. Todo este proceso, si se tiene que hacer sin usar agentes, puede llevar varios días, pero aquí estaríamos optimizando el tiempo, ya que se haría en cuestión de minutos.

GPT-4 es capaz de jugar al MINECRAFT de forma AUTONOMA:

Como fue charlado en el curso, quiero dejar plasmado Voyager, el trabajo de NVIDIA que implementa un agente autónomo potenciado por un modelo del lenguaje, que es capaz de jugar muy bien al Minecraft.



Este sería el ciclo que tiene la inteligencia artificial para ser autónoma. Aprovechando su conocimiento, puede organizar, planificar, ejecutar, criticar, aprender y nuevamente volver a organizar. En la historia, se pueden marcar grandes momentos, y esto es uno de ellos.

<https://www.youtube.com/watch?v=ZwluMd2nAWU>

<https://voyager.minedojo.org/>

<https://github.com/MineDojo/Voyager>

FUENTES:

<https://www.unir.net/ingenieria/revista/agentes-inteligentes/>

<https://la.blogs.nvidia.com/2023/03/29/para-que-se-utilizan-los-grandes-modelos-de-lenguaje/>

<https://gafowler.medium.com/revolutionizing-ai-the-era-of-multi-agent-large-language-models-f70d497f3472>

<https://www.youtube.com/watch?v=Vcrz6zdmQDM>

<https://www.youtube.com/watch?v=ZwluMd2nAWU>

<https://www.youtube.com/watch?v=1bGgeol-nL0&t=20s>

https://www.youtube.com/watch?v=XK8GR2il_4E&t=139s