

## Tomás Navarro Miñón Ejercicio 2

### Ejercicio 2: TP NLP

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.

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

Opcional: Resolución con código de dicho escenario.

### Investigación sobre el Estado del Arte de Aplicaciones de Agentes Inteligentes con LLMs Libre

- Asistentes Virtuales: Aplicaciones como Siri, Google Assistant y Cortana utilizan modelos de lenguaje generativos para entender y responder a las consultas de los usuarios.
- Chatbots Conversacionales: Empresas utilizan LLMs en chatbots para ofrecer soporte al cliente, responder preguntas frecuentes y facilitar interacciones conversacionales.
- Generación Automática de Contenido: Herramientas como OpenAI's GPT-3 se aplican en la generación de contenido creativo, redacción automática y resúmenes de texto

#### Asistentes Virtuales:

Lo que antes era ciencia ficción, hoy se convierte en una realidad al alcance de muchos. En noviembre de 2022 se lanzó ChatGPT, el chatbot gratuito de OpenAI que ha conquistado Internet. ChatGPT utiliza una Inteligencia Artificial basada en los modelos de procesamiento del lenguaje, más concretamente LLM (Large Language Model). La cuarta revolución industrial se extiende, se integra y se dirige a nuestras capacidades cognitivas.

Estos modelos surgidos en 2018 procesan y analizan la información a través de **redes neuronales** para comprender lenguaje y responder preguntas sobre el texto aprendido. Si bien es cierto que fueron entrenados para tareas más sencillas como predecir la siguiente palabra de una frase incompleta, se ha observado que funcionan para una gran variedad de tareas. Esto se debe a que capturan la semántica y sintaxis de nuestro lenguaje y memorizan a su vez datos de la fase de entrenamiento.

Entre los usos más comunes destacan el análisis de sentimientos, la traducción, el resumen o la generación de texto, entre otras, aunque su uso se ha disparado por el desarrollo de **asistentes de IA** (chatbots), especialmente GPT.

Una vez se ha entrenado el modelo, puede realizar dos tareas principales: **predicción y generación de texto**. Cuando le presentamos a un LLM una secuencia de palabras, intenta predecir la siguiente secuencia en función del contexto proporcionado. Por ejemplo, si le damos la frase "Hoy hace buen tiempo, voy a dar un paseo por el...", el modelo puede

predecir que la siguiente palabra será "parque" o "vecindario" en base al contexto; esta es la tarea principal de predicción. Por otro lado, si le damos una frase como "Una vez en un lugar lejano...", el modelo puede continuar la historia generando una secuencia de palabras coherentes y relevantes; esta sería la tarea de generación de texto.

El entrenamiento requiere convertir el texto en una representación numérica que sea capaz de interpretar el modelo, además de los parámetros y una función de pérdida para medir los resultados. El objetivo del modelo será **optimizar** los parámetros para minimizar la pérdida, repitiendo el proceso hasta que el nivel de precisión sea aceptable. El modelo, como se ha mencionado, es una red neuronal que se inspira en el funcionamiento humano. Se puede entender el modelo como una conexión de neuronas en las que cada una de ellas realiza una operación matemática y, a través de estas redes, el LLM toma decisiones sobre frases o palabras en función de la probabilidad en un contexto determinado.

Pero cabe destacar que los LLM no comprenden el texto ni tienen un conocimiento real del mundo: no entienden el significado de las palabras ni poseen conciencia como un ser humano. Asimismo, no son capaces de sentir ni empatizar; simplemente aplican patrones estadísticos y reglas aprendidas para generar un texto coherente prediciendo la siguiente palabra.

Vivimos en un mundo de constantes cambios, más si cabe en el mundo de la tecnología. Según las estadísticas, el 80% de los usuarios de internet han interactuado en alguna ocasión con un chatbot. Mientras que los primeros asistentes se limitaban a responder preguntas con respuestas predefinidas, hemos experimentado como en los últimos meses estas conversaciones son más elaboradas y naturales gracias al uso de los modelos LLM. De modo que, aunque es hoy habitual preguntarse cómo será nuestro futuro con la Inteligencia Artificial, **ya forma parte de nuestro día a día.**

## Chatbots:

A medida que continúa el desarrollo de grandes modelos de lenguaje, existe un gran interés en crear agentes de IA autónomos. AutoGPT, GPT-Engineer y BabyAGI son ejemplos de agentes autónomos impulsados por LLM que desarrollan un LLM subyacente para comprender el objetivo que se les ha asignado y trabajar para lograrlo. El agente recuerda hasta dónde ha progresado, utiliza el LLM para razonar sobre qué hacer a continuación, toma acciones y comprende cuándo se ha cumplido el objetivo. Esto a menudo se conoce como razonamiento en cadena de pensamientos y, de hecho, puede funcionar. Uno de nuestros equipos implementó un chatbot de atención al cliente como agente autónomo. Si el chatbot no puede lograr el objetivo del cliente, reconoce su propia limitación y, en su lugar, redirige al cliente hacia un humano. Este enfoque definitivamente se encuentra en una etapa temprana de su ciclo de desarrollo: los agentes autónomos a menudo sufren una alta tasa de fallas e inciden en costosas tarifas de servicios de IA, y al menos una startup de IA se ha alejado del enfoque basado en agentes.

Agentes Autónomos basados en LLMs actuales para generación automática

Auto-GPT (147k ★) • Un agente de inteligencia artificial que puede realizar diversas tareas de forma autónoma utilizando Internet y otras herramientas. Está basado en GPT-4 de OpenAI, un modelo de lenguaje grande que puede generar texto para diferentes propósitos.

GPT-Engineer (42k ★) • Un proyecto que utiliza GPT-4 para automatizar el proceso de ingeniería de software. Incluye varios scripts en Python que interactúan con el modelo GPT-4 para generar código, aclarar requisitos, generar especificaciones y más.

AgentGPT (26k ★) • Puedes asignarle simplemente un nombre y un objetivo a este agente de inteligencia artificial, y él pensará en las mejores tareas para realizar, las ejecutará, evaluará los resultados y repetirá el proceso hasta alcanzar el objetivo.

MetaGPT (23.3k ★) • Toma un requisito de una línea como entrada y produce varios artefactos, como historias de usuario, análisis competitivo, requisitos, estructuras de datos, APIs, documentos y código.

BabyAGI (16.7k ★) • Dado un objetivo específico y un punto de partida (tarea inicial), genera ideas creativas, las organiza en tareas y prioriza esas tareas para guiarte hacia la consecución de tu objetivo

JARVIS (21.7k ★) • Un sistema que conecta LLMs con varios modelos de aprendizaje automático para resolver tareas de inteligencia artificial complejas. JARVIS se inspira en el asistente de inteligencia artificial ficticio de Iron Man y utiliza el lenguaje natural como una interfaz universal para comunicarse con usuarios y modelos.

SuperAGI (11.8k ★) • Puedes ejecutar agentes concurrentes de manera fluida, ampliar las capacidades del agente con herramientas. Los agentes realizan eficientemente una variedad de tareas y mejoran continuamente su rendimiento con cada ejecución subsiguiente.

GPT Researcher (4k ★) • Un agente autónomo que puede producir informes de investigación detallados, objetivos e imparciales, con opciones de personalización para centrarse en recursos relevantes, esquemas y lecciones.

ShortGPT (3.5k ★) • Puede generar contenido breve y creativo, como chistes, eslóganes, titulares, tweets y más.

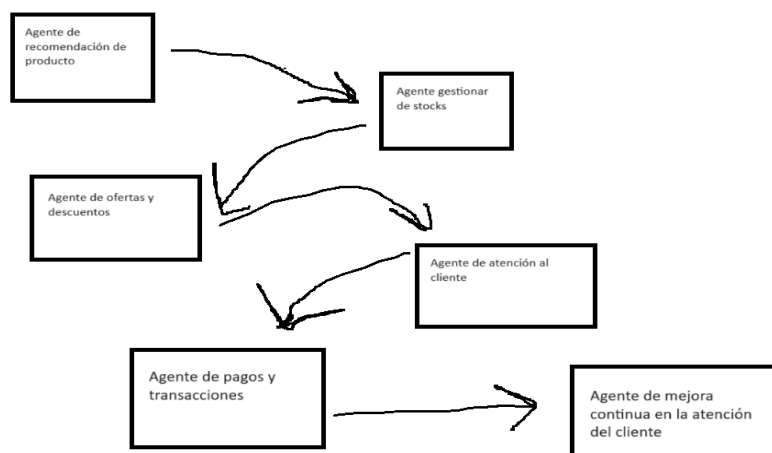
MiniAGI (2.5k ★) • Un agente autónomo general de propósito mínimo basado en GPT-3.5 / GPT-4. Puede analizar precios de acciones, realizar pruebas de seguridad de red, crear arte y ordenar pizza.

## Problemática a Solucionar:

Mejorar el problema de la experiencia de las compras en un shopping.

1. Agente de recomendación de producto: Utilizar historiales de compra, cookies y preferencias del usuario para recomendar productos de manera personalizada. Este se adaptaría e iría aprendiendo continuamente a los cambios en los gustos del cliente.
2. Agente gestor de stocks: Este agente controlaría los niveles de stock en las tiendas y notificaría automáticamente a los equipos de reposición cuando sea necesario reabastecer productos populares.
3. Agente de ofertas y descuentos: Analizaría patrones de compra y tendencias del mercado para generar ofertas y descuentos al público en general, tendría en cuenta cuestiones como fechas importantes, feriados, el clima, etc . Aumentaría la satisfacción del cliente al proporcionar precios atractivos.
4. Agente de atención al cliente: Respondería preguntas, proporcionaría información sobre productos y manejaría problemas de servicio al cliente. Además podría resolver dudas a los mismos, sería capaz de poder concretar una venta si está bien entrenado. Ya para una funcionalidad mayor podría derivar a los usuarios a agentes especializados según sea necesario.
5. Agente de pagos y transacciones: Facilita transacciones sin problemas, permitiendo a los clientes realizar pagos de forma rápida y segura. Interactuaría con métodos de pago digitales y criptomonedas.
6. Agente de mejora continua en la atención del cliente: Recopilara comentarios de los clientes sobre su experiencia de compra. Utiliza esta información para mejorar servicio y ajustar las recomendaciones de productos.

## Esquema del sistema multiagente:



Fuentes:

Fuente: <https://byby.dev/ai-autonomous-agents>

Fuente: <https://www.raona.com/asistentes-virtuales-asi-es-como-ayudan-en-nuestro-dia-a-dia-dentro-y-fuera-de-la-empresa/>

Fuente: <https://es.linkedin.com/pulse/llm-el-futuro-de-los-asistentes-virtuales>

Fuente: Investigación y conocimiento propio