Glosario de la IA

A continuación, presento un glosario de términos técnicos fundamentales de Inteligencia Artificial basado en las fuentes proporcionadas:

Conceptos Fundamentales

- **LLM (Large Language Model):** Es un modelo de lenguaje a gran escala consistente en una red neuronal entrenada para **predecir el siguiente token** dado un contexto determinado. Ejemplos conocidos son GPT, Claude y Gemini.
- **Token:** Es la unidad básica de procesamiento de un LLM. No equivale necesariamente a una palabra o letra; puede ser una palabra completa, una parte de ella, un número o un signo de puntuación. Su manejo es clave para entender los costos y los límites de memoria de los modelos.
- **Vector / Embedding:** Es la representación numérica del significado de un token en un espacio matemático. Los tokens con significados similares quedan "cerca" geográficamente en este mapa de múltiples dimensiones, lo que permite realizar **búsquedas semánticas** por proximidad en lugar de palabras exactas.
- **Attention (Atención):** Mecanismo que permite al modelo mirar diferentes partes de un texto para determinar el significado de un token basándose en las palabras que lo rodean. Esto resuelve ambigüedades (como distinguir entre un "banco" de plaza o uno financiero).
- **Transformer:** Es la arquitectura interna de red neuronal que utilizan casi todos los LLM actuales. Se basa en capas apiladas que transforman secuencialmente la información para realizar inferencias cada vez más abstractas.
- **Parámetros:** Son números internos (comparados con "perillas") que se ajustan durante el entrenamiento. Indican la capacidad del modelo: un modelo de "70B" tiene 70.000 millones de parámetros ajustados para generar respuestas precisas.

Entrenamiento y Configuración

- **Pretraining (Preentrenamiento):** Primera fase donde el modelo aprende sobre el lenguaje y el mundo leyendo cantidades masivas de texto de internet, libros y código.
- **Fine-tuning:** Fase de ajuste fino donde un modelo base se entrena con ejemplos específicos de "instrucción-respuesta" para que aprenda a comportarse como un asistente o se especialice en un dominio particular.
- **RLHF (Reinforcement Learning from Human Feedback):** Aprendizaje por refuerzo con retroalimentación humana. Se basa en que personas reales evalúan qué respuestas del modelo son mejores, ayudándolo a ser más útil y seguro.
- **Context Window (Ventana de contexto):** El límite máximo de tokens que el modelo puede procesar en una sola llamada, incluyendo instrucciones, historial y documentos adjuntos.
- **System Prompt:** Una capa de instrucciones invisibles para el usuario que define el rol, comportamiento y reglas que el modelo debe seguir en cada interacción.
- **Temperatura:** Parámetro que controla la aleatoriedad de las respuestas. En **0** el modelo es determinista (siempre elige lo más probable), mientras que valores cercanos a **2** fomentan la creatividad y diversidad.

Estrategias de Uso y Arquitecturas Avanzadas

- **Prompt Engineering:** El arte de construir el contexto adecuado para obtener el mejor resultado posible del modelo.
- **Few-Shot Prompting:** Técnica que consiste en mostrarle al modelo algunos ejemplos de la tarea deseada dentro del prompt para que infiera el patrón.

- **Chain of Thought (Cadena de pensamiento):** Pedirle al modelo que "piense paso a paso" antes de responder, lo que mejora significativamente su precisión en tareas lógicas complejas.
- **RAG (Retrieval Augmented Generation):** Método que permite al modelo acceder a información externa (documentos de una empresa, bases de datos) en el momento de la consulta sin necesidad de reentrenarlo.
- **Vector Database:** Base de datos especializada en almacenar y buscar vectores de manera eficiente por similitud semántica.
- **MCP (Model Context Protocol):** Protocolo estándar que permite conectar modelos de IA con herramientas y fuentes de datos externas (como Slack, GitHub o bases de datos) de forma universal.
- **Agente:** Un sistema autónomo donde el LLM actúa como "cerebro" que decide qué herramientas usar y qué pasos seguir para completar una tarea compleja.
- **Skill:** Paquete de conocimiento específico que se le entrega a un agente para que lo active solo cuando la tarea lo requiera.

Optimización y Nuevas Capacidades

- **Cuantización:** Técnica para comprimir modelos reduciendo la precisión de sus parámetros (de 32 bits a 8 o 4 bits), permitiendo que modelos grandes corran en computadoras hogareñas.
- **Destilación:** Crear un modelo pequeño (estudiante) que imita el comportamiento de uno más grande y capaz (maestro), resultando en un sistema más rápido y liviano.
- **Multimodal:** Capacidad de un modelo para procesar y generar no solo texto, sino también imágenes, audio y video
- **Context Engineering (Ingeniería de Contexto):** Es el siguiente nivel del *prompt engineering*. Consiste en gestionar de forma dinámica y eficiente qué información le llega al modelo a lo largo del tiempo para no exceder el límite de la **ventana de contexto**. Utiliza técnicas como:
    - **Sliding window:** Enviar solo los últimos mensajes y resumir el historial anterior.
    - **Sumarización dinámica:** Usar modelos más baratos para compactar información no crítica.
    - **RAG selectivo:** Inyectar solo fragmentos de conocimiento relevantes para la consulta actual.
- **Hooks:** Son mecanismos que permiten automatizar el comportamiento de un **agente** en momentos específicos de su flujo de trabajo. Los más comunes son:
    - **Pretool hook:** Se ejecuta antes de usar una herramienta para validar permisos o registrar acciones.
    - **Post-tool hook:** Se activa tras recibir el resultado de una herramienta para validar o transformar el *output*.
    - **Precommit hook:** Realiza acciones (como ejecutar tests) antes de que el agente confirme un cambio.
- **Modelos Locales:** Son modelos de código abierto (como Llama, Mistral o Gemma) que se descargan y ejecutan directamente en la computadora del usuario en lugar de la nube. Proyectos como **Ollama** facilitan este proceso. Sus principales ventajas son el **costo cero**, la **privacidad total** (los datos nunca salen de la máquina) y la falta de latencia de red, aunque su rendimiento depende totalmente del hardware (GPU y RAM) disponible.
- **Cuantización (Quantization):** Técnica de compresión que permite reducir drásticamente el uso de memoria de un modelo al representar sus **parámetros** con menos bits (por ejemplo, pasar de 32 bits a 8 o 4 bits). Esto hace posible que modelos

enormes, que originalmente requerirían hardware de centros de datos, puedan ejecutarse en computadoras personales con una pérdida de calidad mínima para la mayoría de las tareas. Se diferencia de la **destilación** en que la cuantización comprime el modelo original, mientras que la destilación entrena un modelo nuevo más pequeño desde cero.

-