2.3 Modelos de Lenguaje y Procesamiento del Lenguaje Natural

Los modelos de lenguaje: cómo las máquinas aprenden a entender y generar texto

En los últimos años, uno de los avances más visibles dentro de la inteligencia artificial ha sido el desarrollo de los **modelos de lenguaje**. Son las tecnologías que permiten que hoy podamos conversar con sistemas de IA, pedirles que redacten textos, expliquen conceptos, generen código o resuman documentos.

Desde fuera puede parecer que estos sistemas **piensan o razonan**, pero en realidad su funcionamiento se basa en una idea bastante simple: **aprender patrones del lenguaje a partir de enormes cantidades de texto**. Los modelos analizan millones o miles de millones de frases y aprenden qué palabras suelen aparecer juntas y en qué contexto.

Para entenderlo de forma sencilla, podemos imaginarlo como cuando una persona ha leído miles de libros y conversaciones: poco a poco empieza a reconocer **cómo se construyen las frases, cómo se relacionan los conceptos y qué tipo de respuestas suelen aparecer en cada situación**.

Cómo se entrena un modelo de lenguaje

Detrás de estos sistemas existe un proceso técnico complejo. Los modelos de lenguaje modernos se entrenan utilizando **redes neuronales muy grandes** con millones o incluso miles de millones de parámetros. Durante el entrenamiento, el modelo aprende una tarea muy concreta: **predecir la siguiente palabra (o token) dentro de una frase**.

Por ejemplo, si el modelo recibe una frase como:

> “La fotosíntesis es el proceso mediante el cual las plantas…”

el sistema calcula qué palabras tienen más probabilidad de aparecer a continuación: *producen*, *generan*, *transforman*, etc.

Al repetir este proceso millones de veces con grandes conjuntos de texto, el modelo acaba aprendiendo:

- gramática y estructura del lenguaje
- relaciones entre conceptos
- patrones comunes de razonamiento
- conocimiento general presente en los datos

Cuando el entrenamiento termina, el modelo puede **generar texto nuevo** utilizando esos patrones aprendidos.

Las etapas para construir un sistema conversacional

El desarrollo de un sistema conversacional basado en IA suele implicar varias fases.

1. Recopilación de datos

El primer paso consiste en reunir un gran conjunto de textos que sirva para entrenar el modelo. Estos datos suelen proceder de muchas fuentes distintas, como:

- libros
- artículos científicos
- páginas web
- documentación técnica
- conversaciones

En modelos orientados a un ámbito específico —por ejemplo, educación científica— los datos pueden incluir explicaciones de biología, problemas resueltos de física o ejercicios de matemáticas.

La calidad de los datos es crucial, porque el modelo **aprende directamente de esos ejemplos.**

2. Preparación del texto

Antes de entrenar el modelo, los datos deben prepararse. Esto implica limpiar los textos, eliminar duplicados y normalizar formatos.

Después se realiza un proceso llamado **tokenización**, en el que el texto se divide en unidades más pequeñas llamadas *tokens*. Un token puede ser una palabra, parte de una palabra o incluso un símbolo.

Por ejemplo, una frase como:

> “La inteligencia artificial aprende rápido”

se transforma en una secuencia de tokens que el modelo puede procesar numéricamente

3. Entrenamiento del modelo

Una vez preparados los datos, comienza el entrenamiento. El modelo recibe una secuencia de tokens y debe **predecir cuál será el siguiente**. Si se equivoca, el algoritmo ajusta los parámetros de la red neuronal para mejorar la predicción.

Este proceso se repite millones de veces. Con el tiempo, el modelo aprende patrones cada vez más complejos del lenguaje.

Los modelos actuales suelen basarse en la arquitectura **Transformer**, que utiliza mecanismos de atención para analizar las relaciones entre palabras dentro de una frase y comprender mejor el contexto

4. Ajuste para conversación

Un modelo entrenado con texto general no necesariamente sabe mantener una conversación. Por eso suele realizarse una fase adicional de ajuste en la que el sistema aprende a responder preguntas o a interactuar con usuarios.

Para ello se utilizan ejemplos de diálogo o pares de **pregunta–respuesta**, como:

Pregunta:
¿Qué es la ley de Ohm?

Respuesta:
La ley de Ohm establece que la intensidad de corriente que circula por un conductor es proporcional al voltaje aplicado…

Este proceso enseña al modelo a generar respuestas más estructuradas y útiles

5. Evaluación y mejora

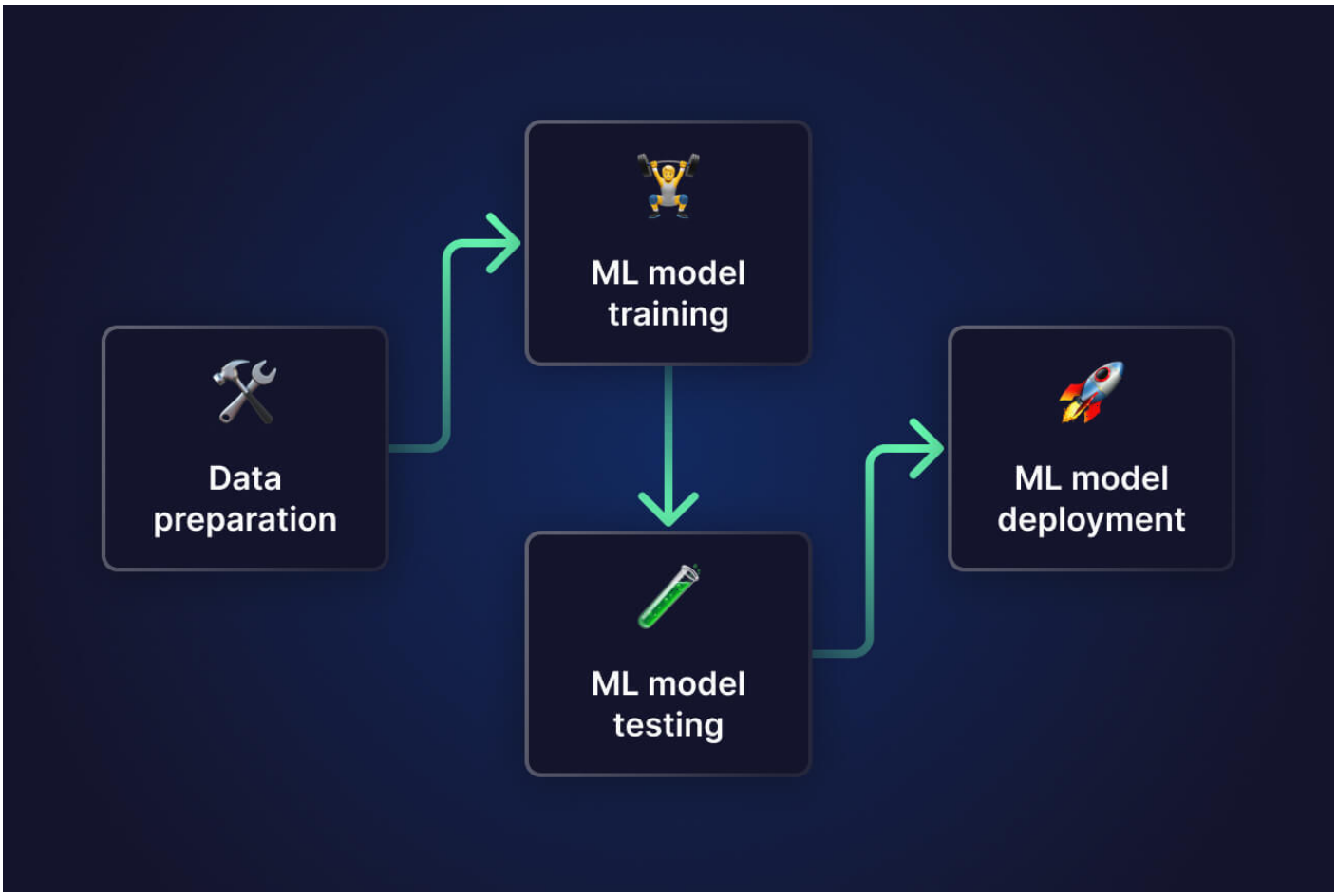
Una vez entrenado, el modelo se somete a diferentes pruebas para comprobar si:

- responde correctamente a preguntas
- mantiene coherencia en conversaciones

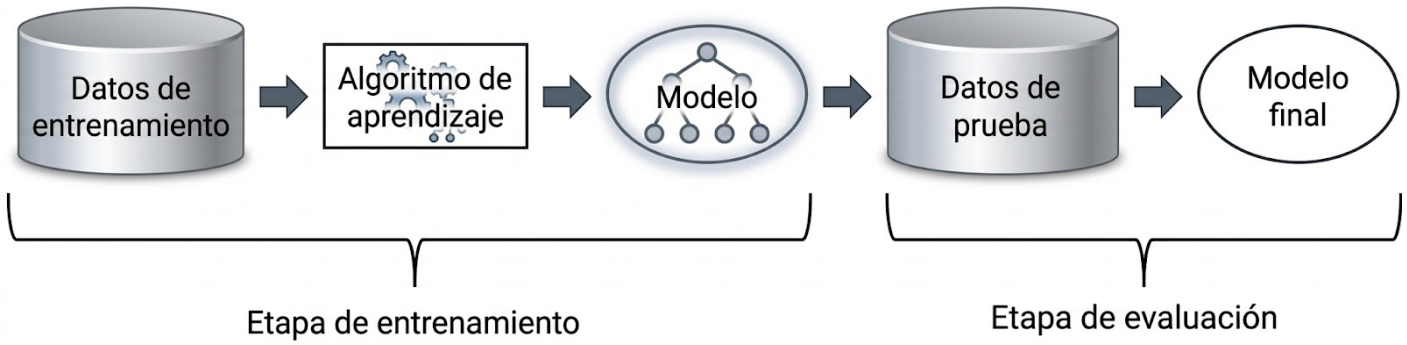
- evita generar información incorrecta

En función de los resultados, se pueden realizar nuevos ajustes o mejoras en el entrenamiento.

Al final del proceso siempre se obtiene un modelo que es la pieza clave de todo puesto que nos va a permitir realizar el proceso de generación de contenido y aplicar las ténicas que veremos después (prompting, RAG y fine tuning)



Esquema de aprendizaje en ML, se entrena al modelo con datos y luego se testea con datos nuevos para valorar su eficacia



Otra versión del esquema anterior. Al final siempre buscamos la obtención de un modelo

NLP, LLM y el auge de la IA generativa

El **Procesamiento del Lenguaje Natural (NLP)** es el área de la inteligencia artificial que intenta que las máquinas puedan **comprender, analizar y generar lenguaje humano**. Durante muchos años, las técnicas de NLP se basaban en reglas lingüísticas o modelos estadísticos relativamente simples. Estos métodos permitían realizar tareas concretas, como clasificar textos o detectar palabras clave, pero tenían limitaciones importantes cuando se trataba de comprender el contexto completo de una frase o manejar conversaciones complejas.

El panorama cambió radicalmente con la llegada del **Deep Learning** y, especialmente, con la arquitectura **transformer**, que permitió construir modelos capaces de analizar grandes cantidades de texto teniendo en cuenta el contexto completo de las palabras. A partir de estos avances surgieron los **Large Language Models (LLM)**, modelos de lenguaje entrenados con enormes colecciones de texto que pueden realizar múltiples tareas lingüísticas con un único sistema.

Gracias a estos modelos, muchas tareas clásicas de NLP pueden abordarse hoy de forma **más eficiente y flexible**. En lugar de construir un sistema diferente para cada tarea —por ejemplo, uno para traducir textos y otro para resumir documentos— los LLM pueden resolver muchas de estas tareas simplemente mediante instrucciones o ejemplos.

Entre las **principales tareas del NLP** encontramos varias que hoy se utilizan de forma cotidiana en múltiples aplicaciones tecnológicas:

- **Traducción automática**, que permite traducir textos entre diferentes idiomas.
- **Análisis de sentimiento**, utilizado para detectar opiniones positivas o negativas en redes sociales o reseñas de productos.
- **Clasificación de textos**, por ejemplo para detectar spam o categorizar documentos.
- **Reconocimiento de entidades**, que identifica nombres de personas, lugares o organizaciones dentro de un texto.
- **Resumen automático de documentos**, útil para procesar grandes cantidades de información.

- **Sistemas de pregunta-respuesta**, capaces de responder preguntas sobre un texto o una base de conocimiento.
- **Conversación automática**, como la que se produce en chatbots y asistentes virtuales.

A partir de estas capacidades del NLP han surgido muchas aplicaciones de **IA generativa**, donde los modelos no solo analizan información, sino que también crean contenido nuevo. Los modelos actuales pueden generar textos completos, redactar artículos, producir código de programación, crear resúmenes o mantener conversaciones relativamente complejas con los usuarios.

Esto ha dado lugar a una gran variedad de herramientas basadas en modelos de lenguaje: asistentes conversacionales, sistemas de apoyo a la programación, motores de búsqueda inteligentes o plataformas educativas capaces de generar explicaciones y ejercicios. Además, estos modelos también se utilizan para generar otros tipos de contenido, como imágenes, audio o vídeo, combinando el lenguaje con otros tipos de datos.

En definitiva, el NLP ha pasado de ser un campo especializado dentro de la inteligencia artificial a convertirse en **uno de los motores principales de la IA actual**. Los avances en Deep Learning y en los modelos de lenguaje han permitido que las máquinas interactúen con el lenguaje humano de una forma mucho más natural, lo que explica el rápido crecimiento de aplicaciones basadas en **chatbots, asistentes inteligentes y sistemas generativos**.

Ampliando los modelos

En la práctica, cuando se quiere construir un sistema conversacional o un *chatbot* basado en inteligencia artificial, existen dos caminos principales. El primero sería entrenar un modelo completamente desde cero, lo que implica construir la red neuronal y alimentarla con enormes cantidades de datos para que aprenda el lenguaje. Este proceso requiere infraestructuras muy potentes, grandes centros de datos y semanas o meses de entrenamiento, por lo que normalmente solo está al alcance de grandes empresas tecnológicas o centros de investigación.

Por esta razón, lo más habitual hoy en día no es empezar desde cero, sino **partir de un modelo ya entrenado y adaptarlo a una tarea concreta**. Los grandes modelos de lenguaje ya poseen un conocimiento general del lenguaje porque han sido entrenados con enormes colecciones de texto. A partir de ahí se pueden ajustar o especializar mediante distintas técnicas.

Una de las más conocidas es el **fine-tuning**, que consiste en volver a entrenar el modelo con un conjunto de datos más pequeño y específico para que aprenda el vocabulario, el estilo o los patrones de un determinado dominio. Por ejemplo, un modelo general puede adaptarse para responder preguntas médicas, jurídicas o educativas entrenándolo con ejemplos propios de ese ámbito. En este proceso se ajustan los parámetros internos del modelo para que responda mejor en ese contexto concreto.

Otra técnica muy utilizada es **RAG (Retrieval-Augmented Generation)**, que en lugar de modificar el modelo permite conectarlo a fuentes de información externas, como bases de datos o colecciones de documentos. Cuando el usuario hace una pregunta, el sistema primero busca información relevante en esos documentos y luego utiliza el modelo de lenguaje para generar la respuesta combinando su conocimiento previo con esos datos recuperados. De esta forma el modelo puede trabajar con información actualizada o especializada sin necesidad de volver a entrenarlo.

También se utilizan estrategias de **prompting avanzado**, que consisten en diseñar cuidadosamente las instrucciones o ejemplos que se proporcionan al modelo para guiar su comportamiento. En muchos casos, una buena forma de plantear la pregunta o proporcionar contexto adicional puede mejorar notablemente la calidad de las respuestas.

En definitiva, los modelos de lenguaje actuales no funcionan como una mente humana que razona de forma consciente. Su comportamiento se basa en **aprender patrones estadísticos del lenguaje a partir de grandes cantidades de datos** y utilizar esos patrones para generar nuevas frases. Sin embargo, cuando estos modelos se entrenan a gran escala y se combinan con técnicas como el fine-tuning o el RAG, pueden producir respuestas sorprendentemente coherentes y útiles.

Para visualizarlo con un ejemplo educativo, imaginemos que queremos construir un modelo orientado a resolver problemas de física de bachillerato. El sistema podría entrenarse con miles de ejemplos donde aparece una pregunta y su resolución paso a paso. Por ejemplo:

“Un objeto de 5 kg acelera a 2 m/s². Calcula la fuerza aplicada.”
El modelo aprendería que debe aplicar la segunda ley de Newton, usar la fórmula **F = m · a** y calcular el resultado.

Del mismo modo, podría entrenarse con ejemplos de biología —como preguntas sobre la función del ADN— o de química, como ajustar ecuaciones químicas. Con suficientes ejemplos, el modelo acaba aprendiendo **los patrones de explicación y resolución de problemas** que aparecen en esos campos.

Esto explica por qué hoy es posible crear asistentes especializados en educación, ciencia o cualquier otro ámbito: no porque la máquina “entienda” el conocimiento como lo haría una persona, sino porque ha aprendido **cómo suelen formularse las preguntas y cómo suelen construirse las respuestas** dentro de esos dominios.

La evolución de los modelos de lenguaje

Durante muchos años los **modelos de lenguaje** eran relativamente simples. Funcionaban con métodos estadísticos que analizaban secuencias cortas de palabras para calcular probabilidades. Por ejemplo, podían estimar que después de la expresión *"buenos"* es muy probable que aparezca *"días"*, o que después de *"por favor"* suele venir *"gracias"*. Estos sistemas eran útiles para tareas básicas como corrección automática o predicción de palabras, pero tenían una limitación importante: **solo podían manejar contextos muy pequeños** y apenas entendían el significado global de una frase.

El gran salto llegó con el desarrollo del **Deep Learning** aplicado al lenguaje y, sobre todo, con la aparición en 2017 de la arquitectura **Transformer**, presentada en el famoso artículo *Attention is All You Need*. Esta arquitectura introdujo el mecanismo de **atención**, que permite a los modelos analizar relaciones entre palabras dentro de una frase completa e incluso entre frases muy separadas dentro de un texto. Gracias a esta innovación, los sistemas podían captar mejor el contexto y procesar el lenguaje de forma mucho más eficiente que los modelos anteriores basados en redes recurrentes.

A partir de ese momento comenzaron a desarrollarse los llamados **Large Language Models (LLM)** o modelos de lenguaje de gran tamaño. Estos modelos utilizan redes neuronales profundas entrenadas con enormes cantidades de texto procedente de libros, páginas web, artículos científicos o conversaciones. Su objetivo es aprender los patrones del lenguaje para poder **comprender y generar texto coherente**.

Uno de los hitos importantes fue el lanzamiento de la familia **GPT** de OpenAI. El primer modelo, GPT-1, apareció en 2018 con unos 117 millones de parámetros. Poco después llegó **GPT-2**, que ya alcanzaba alrededor de 1.500 millones de parámetros. En 2020 se presentó **GPT-3**, con aproximadamente 175.000 millones de parámetros, lo que permitió generar textos sorprendentemente coherentes y realizar múltiples tareas lingüísticas con un mismo modelo.

En paralelo surgieron otros modelos importantes. Por ejemplo, **BERT**, desarrollado por Google, se centró en mejorar la comprensión del lenguaje utilizando representaciones bidireccionales del contexto. Este modelo se convirtió en uno de los más influyentes en tareas de NLP como clasificación de textos o sistemas de pregunta-respuesta.

Con el tiempo, la investigación en modelos de lenguaje se aceleró enormemente. En la actualidad existen numerosos LLM desarrollados tanto por grandes empresas tecnológicas como por comunidades de investigación abiertas.

Entre los más conocidos podemos mencionar:

GPT (OpenAI)
La familia GPT (Generative Pre-trained Transformer) es probablemente la más popular. Estos modelos han impulsado el auge reciente de los asistentes conversacionales y de muchas herramientas de IA generativa.

Gemini (Google)
Es la evolución de los modelos de lenguaje desarrollados por Google. Está diseñado para trabajar de forma multimodal, combinando texto, imágenes y otros tipos de información.

Claude (Anthropic)
Este modelo se ha diseñado poniendo especial énfasis en la seguridad y el alineamiento con valores humanos, intentando reducir riesgos asociados al uso de la inteligencia artificial.

Llama (Meta)
Una de las familias de modelos más influyentes en el ecosistema open source. Varias versiones han sido liberadas públicamente, lo que ha permitido a investigadores y desarrolladores crear nuevas aplicaciones basadas en ellos.

Mistral
Un proyecto europeo que ha ganado relevancia por desarrollar modelos relativamente eficientes, capaces de ofrecer buen rendimiento incluso en hardware más modesto.

Qwen (Alibaba)
Una familia de modelos que ha demostrado un rendimiento competitivo en múltiples idiomas y que también cuenta con versiones accesibles para uso local.

Aunque estos modelos comparten una base tecnológica común —los **transformers**— pueden diferir mucho en distintos aspectos: el tamaño del modelo, los datos utilizados para entrenarlo, las optimizaciones internas o las licencias de uso.

En conjunto, la evolución de los modelos de lenguaje ha sido extraordinariamente rápida. En apenas una década se ha pasado de sistemas capaces de completar frases simples a modelos que pueden **mantener conversaciones complejas, explicar conceptos científicos, generar código o analizar grandes cantidades de información**. Esta evolución ha sido uno de los factores clave que han impulsado el desarrollo de la **IA generativa moderna** y la proliferación de asistentes inteligentes en múltiples ámbitos.

Parametrización de modelos

Para comprender cómo funcionan realmente los **modelos de lenguaje actuales**, conviene conocer algunos conceptos fundamentales. Estos conceptos explican cómo procesan el texto, cómo se entrenan, qué recursos necesitan y por qué algunos modelos son enormes mientras otros

pueden ejecutarse en un ordenador personal

1. Tokens: las unidades básicas del lenguaje

Los modelos de lenguaje no trabajan directamente con palabras completas como hacemos los humanos. En su lugar, el texto se divide en pequeñas unidades llamadas **tokens**.

Un token puede ser:

- una palabra completa
- parte de una palabra
- un número
- un signo de puntuación

Por ejemplo, una palabra larga como *“computadora”* puede dividirse en varios tokens dependiendo del sistema de tokenización utilizado.

Los tokens son importantes por varias razones:

- determinan **cuánta información puede procesar el modelo**
- influyen en **el coste de uso de muchos servicios de IA**
- marcan **la longitud máxima de una conversación**

Los modelos generan texto **prediciendo el siguiente token más probable** basándose en los tokens anteriores

2. Contexto: la memoria del modelo

El **contexto** (o *context window*) es la cantidad de texto que el modelo puede analizar al mismo tiempo. En otras palabras, es la cantidad de tokens que el modelo puede “recordar” durante una conversación o una tarea.

Cuanto mayor es el contexto, más información puede utilizar el modelo para responder.

Esto es clave en tareas como:

- analizar documentos largos
- resumir informes
- revisar código
- mantener conversaciones complejas

Los modelos antiguos tenían contextos muy pequeños (unos cientos o miles de tokens). Hoy existen modelos capaces de manejar **cientos de miles o incluso millones de tokens**, lo que permite analizar documentos muy extensos o incluso libros completos.

Sin embargo, aumentar el contexto también aumenta el **coste computacional**, porque el cálculo de atención en los transformers crece rápidamente con la longitud del texto

3. Prompt: la instrucción que guía al modelo

El **prompt** es la instrucción o pregunta que el usuario proporciona al modelo.

Puede ser algo simple:

> “Explica qué es la fotosíntesis”

o algo más elaborado:

> “Explica la fotosíntesis para alumnos de 1º de ESO usando ejemplos sencillos”.

La forma en que se formula el prompt influye mucho en la calidad de la respuesta. Por eso en los últimos años ha surgido una disciplina conocida como **ingeniería de prompting**, que estudia cómo diseñar instrucciones eficaces para los modelos

4. Cómo se entrenan los modelos de lenguaje

Los LLM se entrenan utilizando **redes neuronales profundas basadas en transformers** y enormes colecciones de texto.

Durante el entrenamiento el modelo aprende a **predecir el siguiente token en una secuencia**. Por ejemplo:

“La fotosíntesis es el proceso mediante el cual las plantas…”

El modelo aprende que las siguientes palabras más probables pueden ser:

- producen
- generan
- transforman

Este proceso se repite **billones de veces** con grandes conjuntos de datos.

Los datasets utilizados suelen incluir:

- páginas web
- libros
- artículos científicos
- código fuente
- documentos técnicos

Por ejemplo, algunos modelos se han entrenado con **más de un billón de tokens de texto** procedentes de múltiples fuentes públicas

5. Tamaño del modelo: los parámetros

Otro concepto clave es el **número de parámetros**.

Los parámetros son los valores internos que la red neuronal ajusta durante el entrenamiento para aprender patrones.

Algunos ejemplos aproximados:

- GPT-1 → 117 millones de parámetros
- GPT-2 → 1.500 millones
- GPT-3 → 175.000 millones

El aumento del número de parámetros permitió mejoras importantes en la capacidad de los modelos para comprender y generar texto.

Sin embargo, los modelos más grandes requieren **enormes recursos de computación**

6. Coste de entrenamiento y recursos necesarios

Entrenar modelos de lenguaje es extremadamente costoso.

Por ejemplo:

- entrenar un modelo de **13 mil millones de parámetros** puede costar alrededor de **1 millón de dólares** y requerir miles de GPUs funcionando durante semanas.

Entrenar modelos gigantes como GPT-3 puede costar **millones de dólares en infraestructura y energía**.

Por esta razón, la mayoría de organizaciones no entrenan modelos desde cero, sino que **adaptan modelos ya existentes** mediante técnicas como:

- **fine tuning**
- **LoRA**

- **RAG**
- **prompt engineering**

Estas técnicas permiten especializar modelos sin repetir todo el entrenamiento.

7. Tabla resumen de algunos modelos populares

Modelo	Organización	Tipo	Parámetros aproximados	Características
GPT-4 / GPT-4o	OpenAI	Propietario	No público	Muy potente, multimodal
Gemini	Google	Propietario	No público	Multimodal, gran contexto
Claude	Anthropic	Propietario	No público	Contexto muy grande
Llama 3	Meta	Abierto	hasta ~70B	Muy usado en investigación
Mistral	Mistral AI	Abierto	7B–Mixtral	Muy eficiente
Qwen	Alibaba	Abierto / mixto	7B–72B	Multilingüe
Falcon	TII	Abierto	hasta 180B	Muy popular en open source
GPT-J	EleutherAI	Abierto	6B	Uno de los primeros LLM abiertos

Tipos de modelos de lenguaje: propietarios, abiertos, online y locales

Hoy en día los **modelos de lenguaje** pueden clasificarse de varias formas según cómo se distribuyen, cómo se ejecutan y qué grado de acceso tenemos a ellos. Comprender estas diferencias es importante porque determina cómo podemos utilizarlos, qué recursos necesitamos y qué control tenemos sobre los datos.

Una primera distinción importante es entre **modelos propietarios y modelos abiertos**.

Los **modelos propietarios** son desarrollados por grandes empresas tecnológicas que no publican completamente su arquitectura, sus datos de entrenamiento o sus pesos internos. El acceso suele realizarse a través de plataformas online o APIs. Ejemplos conocidos son los modelos **GPT de OpenAI**, **Gemini de Google** o **Claude de Anthropic**. Estos modelos suelen ofrecer un

rendimiento muy alto porque están entrenados con enormes infraestructuras y grandes volúmenes de datos. Sin embargo, su uso depende de las condiciones de la empresa que los desarrolla y normalmente implica acceso a través de servicios en la nube.

Por otro lado, existen los **modelos abiertos u open source**, en los que gran parte del modelo se publica para que investigadores y desarrolladores puedan utilizarlos, estudiarlos o adaptarlos. Ejemplos conocidos son **Llama** (Meta), **Mistral**, **Falcon** o algunas versiones de **Qwen**. Estos modelos han impulsado mucho la investigación porque permiten experimentar, crear nuevas aplicaciones o ejecutar inteligencia artificial sin depender completamente de grandes plataformas tecnológicas.

Otra clasificación muy importante es la forma en la que se ejecutan los modelos: **online o localmente**.

Muchos modelos actuales se utilizan **a través de APIs en la nube**. En este modelo, el usuario o el desarrollador envía una consulta a un servidor a través de internet y recibe la respuesta del modelo. Este enfoque tiene varias ventajas: no requiere disponer de hardware potente, permite acceder a modelos muy grandes y las empresas pueden actualizar continuamente los sistemas. Sin embargo, también implica dependencia de conexión a internet, posibles costes de uso y menor control sobre los datos enviados.

Frente a este modelo han surgido en los últimos años herramientas que permiten ejecutar **modelos de lenguaje directamente en un ordenador local**, sin necesidad de conexión a internet. Plataformas como **Ollama**, **LM Studio** o **text-generation-webui** permiten descargar modelos y utilizarlos de forma privada en el propio equipo. Estas herramientas actúan como gestores que permiten instalar, ejecutar y probar modelos de lenguaje en local.

El uso local tiene varias ventajas importantes. Por un lado, mejora la **privacidad**, ya que las consultas y los documentos analizados no salen del ordenador o del servidor interno. Además, permite **integrar modelos en sistemas propios** o en entornos corporativos sin depender de servicios externos. Por ejemplo, una organización puede cargar documentos internos y crear un sistema de consulta basado en IA sin enviar esa información a servicios en la nube.

Finalmente, también podemos distinguir entre **modelos grandes y modelos ligeros**. Los modelos más grandes pueden tener cientos de miles de millones de parámetros y requieren grandes infraestructuras para funcionar. Son los que suelen utilizarse en servicios en la nube. En cambio, han aparecido versiones más **ligeras o compactas** que sacrifican parte del rendimiento a cambio de poder ejecutarse en ordenadores personales o servidores pequeños. Este tipo de modelos permite experimentar con IA de forma local y accesible, algo especialmente interesante en entornos educativos o de investigación.

En conjunto, el ecosistema actual de modelos de lenguaje es muy diverso. Existen modelos abiertos y propietarios, servicios online y sistemas que funcionan localmente, así como versiones gigantes y versiones ligeras. Esta diversidad es precisamente una de las razones por las que la inteligencia artificial se está extendiendo tan rápidamente: **cada organización puede elegir el tipo de modelo que mejor se adapte a sus necesidades, recursos y nivel de control sobre los datos**.

El siguiente paso en los modelos: los agentes de inteligencia artificial

En los últimos años está empezando a aparecer un nuevo concepto que muchos investigadores consideran el **siguiente paso en la evolución de la inteligencia artificial**: los **agentes de IA**.

Hasta ahora, la mayoría de aplicaciones basadas en modelos de lenguaje funcionan de forma relativamente simple. El usuario hace una pregunta, el modelo analiza el texto y genera una respuesta. Es un proceso muy potente, pero también bastante limitado: el sistema **responde**, pero no **actúa**.

Los **agentes de inteligencia artificial** amplían esa idea. Un agente puede entender una tarea, dividirla en pasos y ejecutar acciones para completarla. En otras palabras, no se limita a generar texto, sino que **planifica, toma decisiones y utiliza herramientas externas para alcanzar un objetivo**.

Esto significa que un agente puede hacer cosas como:

- planificar tareas complejas
- consultar información en diferentes fuentes
- utilizar herramientas externas (APIs, bases de datos, buscadores)
- ejecutar acciones en sistemas digitales

Por ejemplo, imaginemos una tarea como **elaborar un informe sobre el cambio climático**. Un modelo de lenguaje clásico podría explicar el tema si se le pregunta. En cambio, un agente podría:

1. buscar información en internet
2. seleccionar los documentos relevantes
3. analizar los datos encontrados
4. generar un informe estructurado
5. enviarlo automáticamente por correo electrónico

Todo este proceso podría realizarse con mínima intervención humana.

El nuevo paradigma: la orquestación de agentes

A medida que estos sistemas evolucionan, ha surgido una idea todavía más interesante: la **orquestación de agentes**.

En lugar de un único sistema que intenta hacerlo todo, se utilizan **varios agentes especializados que colaboran entre sí**, cada uno con una función concreta. Este enfoque consiste en coordinar diferentes agentes dentro de un mismo sistema para alcanzar un objetivo común.

Por ejemplo, en un sistema más complejo podrían intervenir:

- un agente que **busca información**
- otro que **analiza datos**
- otro que **genera informes**
- otro que **toma decisiones o ejecuta acciones**

De alguna forma, este modelo recuerda al funcionamiento de **un equipo humano de trabajo**, donde cada especialista aporta una capacidad concreta para resolver un problema más grande

Un cambio en la forma de interactuar con la tecnología

Este avance también está cambiando la manera en que interactuamos con los ordenadores. Durante décadas, utilizar un sistema informático implicaba aprender comandos, interfaces complejas o programas especializados.

Con los modelos de lenguaje y los agentes, cada vez más herramientas permiten interactuar **simplemente mediante lenguaje natural**. Además, los agentes pueden actuar de forma más autónoma, realizando tareas completas en lugar de limitarse a responder preguntas.

Esto abre posibilidades interesantes en muchos ámbitos, incluido el educativo. Por ejemplo, podrían aparecer:

- asistentes que ayudan a preparar materiales didácticos
- sistemas que analizan grandes cantidades de información académica
- herramientas de tutoría personalizada para el alumnado
- plataformas que automatizan tareas administrativas o de evaluación

Comprender la tecnología para usarla con criterio

Aunque estas tecnologías son muy potentes, también es importante entender sus **capacidades y sus limitaciones**. Los agentes no sustituyen el juicio humano ni el pensamiento crítico. Son herramientas que pueden ayudar a automatizar tareas y gestionar información, pero su uso requiere supervisión y criterio.

En el ámbito educativo, comprender cómo funcionan los modelos de lenguaje y los agentes es especialmente importante. No solo permite utilizarlos de forma más eficaz, sino también **enseñar a los estudiantes a entender críticamente las tecnologías que están transformando nuestra forma de trabajar y aprender**.

Revision #6
Created 2026-03-16 08:23:17 CET by Luis Hueso
Updated 2026-03-16 19:13:30 CET by Luis Hueso