

Newsletter trimestral
CÁTEDRA
iDANAE

INTELIGENCIA · DATOS · ANÁLISIS · ESTRATEGIA

4T25

Especialización de LLM: hacia una
optimización por tarea y ámbito



POLITÉCNICA

UNIVERSIDAD
POLITÉCNICA
DE MADRID

MSO Management
Solutions
Making things happen

Introducción

A medida que el uso de la inteligencia artificial (IA) avanza desde la fase de experimentación hacia su despliegue real está surgiendo un patrón claro: **los modelos de lenguaje de propósito general (LLM)** no son totalmente suficientes en tareas que requieren **una comprensión profunda de un dominio**. Muchas aplicaciones necesitan modelos capaces de interpretar terminología especializada o razonar con información que no es fácilmente accesible por los modelos de propósito general. Esto resulta especialmente relevante en ámbitos técnicos y especializados, como la **sostenibilidad (ESG)**, **riesgo crediticio**, **análisis legal o el ámbito de la salud**, entre otros, donde la precisión y el contexto son fundamentales.

Esta constatación ha impulsado un creciente interés en los **LLM especializados**: modelos diseñados para destacar dentro de un área de conocimiento definida. La actual ola de innovación refleja un esfuerzo más amplio por **equilibrar la capacidad lingüística general con la precisión específica del dominio**.

Existen **tres enfoques técnicos principales** que se utilizan habitualmente para hacer disponible e incorporar conocimiento experto a los LLM:

1. **Escalado de modelos generales**, aprovechando el amplio conocimiento integrado en los LLM de gran tamaño, que pueden cubrir parcialmente ciertos dominios especializados.
2. **Ajuste fino (*fine-tuning*)**, que consiste en adaptar un modelo preentrenado modificando sus pesos —total o

parcialmente— con datos específicos del dominio, o bien utilizando variantes ya ajustadas.

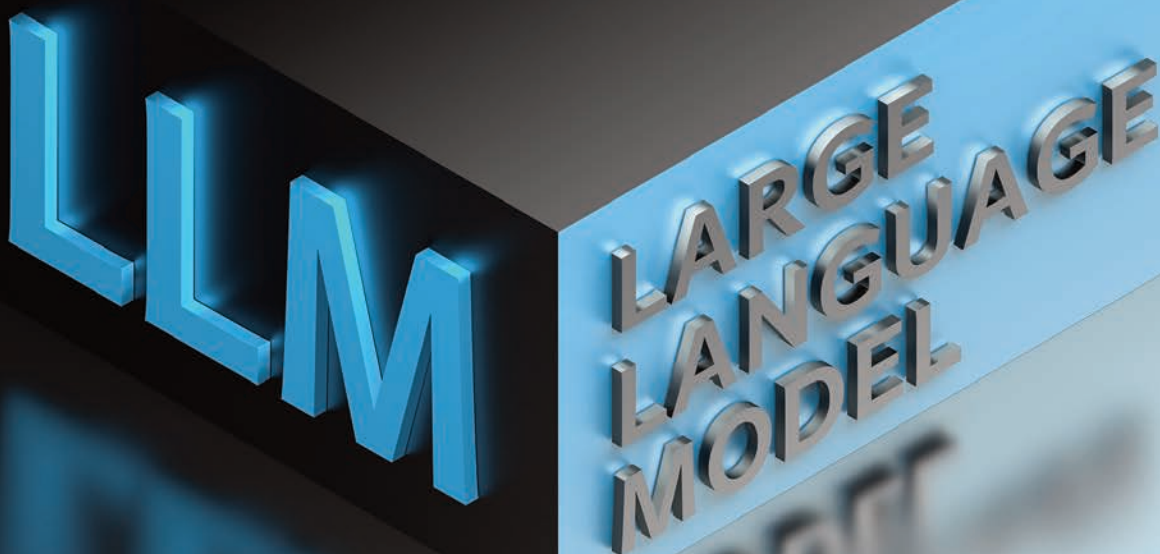
3. **Aumento de conocimiento (*knowledge augmentation*)**, mediante métodos como **RAG (*Retrieval-Augmented Generation*)**, que enriquece las respuestas al recuperar información externa relevante, e **ICL (*In-Context Learning*)**, que permite al modelo inferir patrones o estilos directamente a partir del contenido del prompt.

En la práctica, **estos enfoques suelen combinarse**, ya que cada uno aporta ventajas distintas en términos de adaptabilidad, precisión y eficiencia.

Además, existe la posibilidad de utilizar **herramientas y frameworks integrados**, que ofrecen soluciones listas para crear agentes o flujos de trabajo especializados sin necesidad de reentrenar completamente un modelo.

Por último, **cabe destacar algunas tendencias**, dentro del campo de los domain expert models, tanto en el ámbito académico como empresarial. Ejemplos relevantes son el creciente uso de **SLM (*Small Language Models*)** y **Model Routers**.

En esta publicación se explora cómo los LLM especializados están **transformando el panorama de las aplicaciones de IA**, permitiendo **resultados más fiables y contextualizados** en dominios complejos y con alto contenido de conocimiento.



Revisión Técnica

Fragilidad técnica: abordar los errores y la escalabilidad

Como se ha señalado previamente, existen múltiples estrategias para adaptar un LLM a un campo de conocimiento o tarea concreta. Tradicionalmente, el paradigma dominante ha sido el **bigger is better** [1] [2] basado en la premisa de que los modelos de mayor tamaño ofrecen **una mayor capacidad de generalización** y versatilidad. En efecto, los denominados **frontier models** generalistas suelen presentar un **rendimiento sobresaliente en una amplia variedad de tareas**, lo que en muchos casos **permite su uso directo**, sin necesidad de personalizaciones adicionales.

Sin embargo, incluso entre estos enormes modelos, cada uno destaca en distintas áreas [3], y sigue existiendo mucho **margen de mejora**, sobre todo en **sectores altamente especializados** [4], caracterizados por la escasez de datos públicos o por la compleja interpretación de éstos. Algunos ejemplos de sectores con estas características son el derecho, la medicina, o las finanzas.

Además, los modelos de gran escala suelen implicar **costes elevados**. En muchos escenarios productivos, este enfoque puede resultar poco óptimo, especialmente cuando las tareas son repetitivas o de alcance limitado. Por el contrario, **existen**

modelos más pequeños que, al ser desplegados de forma controlada o mediante soluciones híbridas, **ofrecen un rendimiento comparable**, o incluso mejor [5] **a una fracción del coste** [6]. Así, aunque los modelos de gran tamaño puedan resolver con éxito una amplia gama de tareas, su uso no siempre representa la opción más adecuada desde una perspectiva de optimización de recursos, escalabilidad y en algunos casos incluso de precisión.

Otra razón de peso por la que la especialización de modelos resulta recomendable en numerosos casos es el **riesgo de alucinación** [7], particularmente en aquellos dominios donde los LLM disponen de un conocimiento limitado o muestran una baja confianza en sus predicciones. Este fenómeno, derivado de la naturaleza probabilística del modelo, puede llevar a generar **información incorrecta** o inventada con apariencia de veracidad. En entornos donde la precisión es crítica, este riesgo puede volverse inaceptable [8].

Es por eso que el **fine-tuning** de modelos [9], **ICL** [10] y el **RAG** [11] han ganado mucha relevancia, al hacer que los modelos sean **más eficientes y precisos**. A continuación, se presentarán los fundamentos de cada uno de estos enfoques, seguidos de una comparativa que permitirá comprender en qué situaciones resulta más apropiado aplicar cada uno de ellos.



Fine-tuning

Afinar (*fine-tuning*) un modelo de lenguaje consiste en **entrenar un modelo ya preentrenado** (como GPT-5, Claude Sonnet 4.5, etc.) utilizando datos propios para adaptarlo a un caso de uso concreto.

Cabe preguntarse si sería posible entrenar un modelo experto desde cero, en vez de realizar *fine-tuning*. Si bien es cierto que es en el **pre-entrenamiento** (*pre-training*) cuando un LLM adquiere la mayoría de su conocimiento [12], esta etapa es **difícilmente replicable** en la mayoría de escenarios prácticos, puesto que requiere de una enorme **cantidad de energía, datos e infraestructura** [13]. A modo de referencia, se estima que el entrenamiento de modelos de gran escala como GPT-4 superó los **100 millones de dólares**, según declaraciones de Sam Altman (CEO de OpenAI). Incluso modelos de tamaño considerablemente menor requieren inversiones muy elevadas, con costes de cómputo que oscilan entre 100.000 y 500.000 dólares, sin incluir los gastos asociados al desarrollo, la curación y el almacenamiento de datos [14].

Realizar un **fine-tuning completo**, es decir, modificar la totalidad de los pesos de un modelo, también **es muy costoso**. Según [15], afinar completamente un modelo de la escala de Llama-3 requeriría de sistemas de almacenamiento con varios petabytes de capacidad, interconexiones de memoria de velocidad ultra-alta y centenares o incluso miles de GPUs de última generación. Este tipo de operación no solo resulta **económicamente inviable** para la mayoría de organizaciones, sino también **técnicamente compleja**, ya que exige una orquestación precisa de la comunicación y sincronización entre GPUs, así como una gestión avanzada del pipeline de entrenamiento para evitar cuellos de botella y pérdidas de eficiencia.

Además, al realizar un fine tuning completo, se introduce el riesgo de **catastrophic forgetting** [16] [17], un fenómeno en el cual el modelo pierde parte del conocimiento previamente adquirido al especializarse excesivamente en una nueva tarea, comprometiendo así su versatilidad y capacidad de generalización.

Es por eso que, en el ámbito del fine tuning, surge el concepto de **PEFT (Parameter Efficient Fine Tuning)** [18]. La idea básica detrás de PEFT consiste en mantener congelada la mayoría de los parámetros del modelo base, reentrenando únicamente un pequeño subconjunto de nuevos parámetros que ajustan su comportamiento a la tarea o dominio deseado. Algunos de los ejemplos más populares son LoRA (Low-Rank Adaption) [19], QLoRA [20] y los llamados adapters [21] [22].

Ejemplos de PEFT

LoRA (Low-Rank Adaptation)

En lugar de actualizar todos los parámetros de un LLM, LoRA inyecta pequeñas matrices de bajo rango en las capas del modelo, que actúan como ajustes adicionales sin modificar los pesos originales. IBM lo describe del siguiente modo: "LoRA es una técnica que adapta un modelo grande añadiendo componentes ligeros al original, en lugar de cambiar el modelo entero". Este enfoque permite reducir drásticamente los costes de entrenamiento y el consumo de memoria, logrando un salto significativo en eficiencia sin sacrificar el rendimiento del modelo [23].

QLoRA

Es una variante de LoRA que combina su enfoque de bajo rango con técnicas de cuantización, reduciendo la precisión numérica de los parámetros para disminuir el uso de memoria sin comprometer sustancialmente el rendimiento. Google Cloud recomienda LoRA para velocidad y costo, y además señala que QLoRA consume ~75% menos memoria GPU [24].

Adapters

Los *adapters* son pequeños módulos adicionales insertados en las capas del transformador, diseñados para ajustar el comportamiento del modelo sin modificar sus pesos originales. Este enfoque permite alternar entre tareas o dominios simplemente sustituyendo los módulos correspondientes. Aunque la incorporación de *adapters* introduce cierta latencia adicional durante la inferencia y conlleva un mayor uso de recursos en entrenamiento, puede ofrecer un rendimiento superior en comparación con técnicas más ligeras, dado que se entrena un porcentaje mayor de parámetros [25].

Context Engineering (ICL y RAG)

ICL y RAG son estrategias englobadas dentro del concepto de **context engineering** [26] [27] una disciplina que busca optimizar la cantidad y la forma en que se introduce la información en una llamada a un LLM para obtener la **respuesta más precisa y eficiente posible**. El *context engineering* permite aprovechar al máximo el potencial de un LLM, posibilitándoles realizar tareas que, sin este enfoque, serían imposibles o mucho menos eficientes [28] [29].

Durante los últimos años, este campo ha experimentado una **evolución acelerada**, lo que ha resultado en una proliferación de **disciplinas de investigación especializadas**, entre las que destacan:

- ▶ **Context Retrieval and Generation:** es el ámbito más relevante para el propósito de este documento. Se centra en determinar qué elementos deben seleccionarse y presentarse al modelo, priorizando aquellos más relevantes o informativos [30]. Esto puede orientarse tanto a incorporar conocimiento externo como a inducir comportamientos específicos [31].
 - ▶ **Context Processing:** estudia cómo estructurar e integrar la información de manera que el modelo la procese de forma más eficiente, facilitando decisiones más precisas. Técnicas como GraphRAG hacen uso de un *Context Processing* avanzado [32].
 - ▶ **Context Management:** se enfoca en optimizar el espacio disponible en el prompt mediante estrategias como compresión de texto, vectorización de contexto o selección dinámica de fragmentos, con el fin de equilibrar relevancia y capacidad de memoria [33].
- Las aplicaciones de estos conceptos permiten extraer el máximo rendimiento posible de los modelos con estrategias como las siguientes:
- ▶ **In Context Learning (primarily Prompt Engineering):** el ICL se basa en la capacidad de los LLM para aprender patrones, estilos o comportamientos simplemente a partir del contenido del *prompt*. A nivel práctico, el ICL ha evolucionado desde simples cadenas de ejemplos (*few-shot prompting*), en las que se muestra al LLM cómo debe comportarse, hasta técnicas de *prompt chaining*, *dynamic prompting* o I2CL (*Iterative In Context Learning*) [34] [35].
 - ▶ **RAG:** esta técnica introduce una capa intermedia entre el input del usuario y la generación del modelo, encargada de recuperar información relevante desde una base de conocimiento externa y suministrarla como contexto al LLM. De este modo, el modelo puede razonar sobre información actualizada y específica. Versiones más avanzadas, como GraphRAG [32] o HybridRAG [36], mejoran el proceso mediante técnicas de *Context Processing* para ofrecer respuestas más coherentes y verificables.



Con la llegada del **Model Context Protocol (MCP)** [37], la capacidad de los LLM para invocar herramientas externas, como APIs, bases de datos, calculadoras o navegadores se **ha estandarizado y unificado**. Este avance permite a los modelos razonar de forma más efectiva sobre información que excede su conocimiento interno, y sienta las bases de los **Intelligent Agent Systems**, considerados “*the pinnacle of context learning*” [26].

Según Douwe Kiela, uno de los investigadores que introdujo la técnica RAG y CEO de Contextual AI, MCP complementa a técnicas como RAG al proporcionar un **marco más limpio y eficiente** para la comunicación entre bases de datos y modelos lingüísticos, [38]. Además, se hace posible la utilización de **Multi-Agent-Systems (MAS) para information retrieval**, lo que puede mejorar la precisión de RAG utilizando un pipeline similar, pero agéntico.

Comparación entre las distintas opciones

Una vez revisadas de forma general las principales técnicas que permiten dotar a un modelo de especialización en un dominio concreto (ICL, RAG y fine tuning), se procede a compararlas, dado que cada enfoque presenta **ventajas y limitaciones** dependiendo del caso de uso.

Una analogía ilustrativa a la hora de comparar estas técnicas es la de varios alumnos con diferentes estrategias de estudio frente a un examen en el que se les preguntará sobre un libro [39] [40].

El primer estudiante ha leído y estudiado el libro, por lo que comprende los conceptos y las relaciones entre ellos: representa el **fine-tuning**. El segundo estudiante no ha estudiado, pero durante el examen puede consultar el libro, buscando directamente las respuestas que necesita. Este sería el equivalente a **RAG**. Por último, un tercer estudiante no ha leído el libro, pero ha recibido instrucciones precisas del profesor sobre cómo (aunque no qué) debe responder a las preguntas, este caso representa **ICL**.

Dependiendo de la complejidad y objetivo del examen, cada alumno obtendrá un rendimiento mejor o peor. Por ejemplo, en un examen de historia con fecha concretas, el RAG podría funcionar mejor, mientras que, en un examen de razonamiento matemático, *fine-tuning* sería más adecuado. En un examen sencillo en el que importe la forma de la respuesta, siendo el conocimiento más o menos general, ICL bastaría y no haría falta hacer esfuerzos adicionales. **Cada técnica requiere un tipo de infraestructura y gasto distinto.**

Requerimientos técnicos de cada estrategia

ICL destaca por su **simplicidad, bajo coste de implementación, y flexibilidad**, superando con creces en estos aspectos a las otras dos técnicas [41]. No requiere infraestructura adicional, generación de *embeddings* ni procesos de entrenamiento, lo que la convierte en una opción especialmente atractiva para **casos simples, fases de experimentación, o adaptación rápida a nuevos contextos.**

Sin embargo, **el rendimiento de ICL tiende a decrecer a medida que aumenta la complejidad de la tarea**, y depende en gran medida tanto de la calidad del *prompt* como de la capacidad del modelo para manejar contextos extensos. ICL suele considerarse la **primera estrategia a evaluar** antes de recurrir a técnicas más costosas, ya que permite validar hipótesis y obtener resultados útiles con un esfuerzo mínimo [34].

Al usar **RAG, los requisitos técnicos y operativos son más exigentes** [28]. Es necesario, disponer de una **base documental** de alta calidad, que servirá como fuente de conocimiento para el modelo. Para que el sistema funcione correctamente, debe integrarse una **base de datos vectorial** donde se almacenen las representaciones numéricas de los textos, junto con un pipeline de ingesta y procesamiento capaz de dividir los documentos en fragmentos manejables (*chunks*).

Además, se requiere un **modelo de embeddings** que convierta el texto en vectores semánticos y un mecanismo de recuperación (*retrieval*) que, ante una consulta, identifique los fragmentos más relevantes dentro de la base de conocimiento. Este proceso **introduce inevitablemente cierta latencia**, ya que la búsqueda y recuperación de información requiere tiempo adicional antes de que el modelo pueda generar una respuesta.

Un entorno básico de RAG puede implementarse mediante servicios gestionados [42](API de *embeddings* y base vectorial en la nube), por lo que la **configuración inicial no tiene por qué ser compleja**, aunque sí resulta **más costosa que ICL**. Los principales costes provienen del almacenamiento de datos, la generación de *embeddings* y el incremento del número de tokens procesados en cada consulta.

El **fine-tuning** es la técnica más exigente de las tres, tanto en tiempo como en recursos [43]. Requiere disponer de una gran cantidad de datos de alta calidad, bien etiquetados y representativos del ámbito o tarea concreta que se busca mejorar. Preparar este dataset y asegurarse de que está bien estructurado, limpio y balanceado suele requerir una gran cantidad de tiempo, y tiene un impacto directo en la calidad del resultado final.

Además, el proceso exige una infraestructura de entrenamiento adecuada, generalmente una o varias GPU, dependiendo del tamaño del modelo base [20]. El proceso, además, requiere personal técnico especializado con experiencia en ajuste de hiperparámetros, validación y monitorización del modelo tras su despliegue. Estas tareas son críticas para evitar el sobreajuste y garantizar la estabilidad del rendimiento en producción [44].

Algunos *frontier models* ofrecen la posibilidad de hacer **fine tuning desde la propia API**, aspecto que soluciona muchos de los problemas mencionados hasta ahora, pero el coste de inferencia puede subir [45]. Además, esta funcionalidad **no es común**, y por ejemplo GPT-5 no ha mantenido esta opción, al haber centrado su desarrollo en funcionalidades orientadas a Agentic-AI.

Rendimiento de cada técnica según caso de uso

La habilidad del fine tuning para **añadir nuevos conocimientos** es materia de debate en la literatura actual, con algunos estudios afirmando utilizar esta técnica para enseñar información factual o externa al modelo puede, de hecho, incrementar la probabilidad de generar **alucinaciones** [46].

En líneas generales, si el objetivo es que un LLM pueda utilizar y razonar sobre **datos nuevos o actualizados**, la literatura señala que RAG constituye la opción más adecuada [47], incluso en contextos altamente especializados [48].

Los estudios recientes parecen coincidir en que el *fine-tuning* es más apropiado para enseñar al modelo **"habilidades"**, mientras que el RAG es más potente a la hora de incorporar **nuevos datos al modelo** [49]. Además, RAG cuenta con la ventaja añadida de poder **cambiar dinámicamente** el conocimiento al que el LLM tiene acceso, lo que lo hace especialmente útil en entornos donde el conocimiento cambia con frecuencia. En la

práctica, aunque RAG por sí solo tiende a ser el estándar empresarial [43], bastando para la mayoría de los casos de uso, **la combinación de ambas técnicas puede ofrecer mejores resultados** [39].

Por ejemplo, en el ámbito legal [43], es útil aplicar *fine-tuning* para que un modelo aprenda a comunicarse como un abogado, adoptando su terminología, estilo argumentativo y comprensión de matices lingüísticos específicos. En otras palabras, el modelo desarrolla una "forma de pensar jurídica". Este tipo de competencias no pueden alcanzarse con RAG, ya que **RAG no enseña al modelo a razonar** o internalizar patrones de pensamiento, sino que simplemente le da acceso a información.

Por otro lado, las leyes y normativas propiamente dichas no deberían incorporarse mediante *fine-tuning*, ya que ello requeriría **cantidades masivas de datos y eliminaría la trazabilidad de las respuestas**. Este conocimiento se introduciría con RAG, lo que además ofrece la ventaja de poder actualizar el sistema fácilmente cuando las leyes cambian o quedan obsoletas, sin necesidad de volver a entrenar el modelo.

Así, una vez determinado que ICL no constituye una opción viable para el caso de uso, **si el objetivo es que el LLM adquiera una habilidad**, un comportamiento específico o una comprensión más profunda de ciertos conceptos, la estrategia más adecuada será **PEFT**. Por el contrario, cuando lo que se busca es que el modelo **incorpore o utilice nuevos datos**, resultará más eficiente y escalable emplear **RAG** [43].

Si no se dispone de una cantidad suficiente de datos como para realizar PEFT sin riesgo de overfitting, pero ICL no alcanza la profundidad necesaria para la tarea, existen **técnicas intermedias** que pueden servir como solución, logrando un nivel de adaptación comparable al de otras variantes de *fine-tuning* y superando la precisión y consistencia de ICL [50] [51].

Ejemplos de herramientas y frameworks integrados

Además de los enfoques previamente descritos, es relevante considerar el uso de plataformas y frameworks integrados **as a service**, que permiten tanto el despliegue de modelos especializados como la creación de agentes expertos dentro de entornos empresariales o técnicos específicos. Estas soluciones

facilitan la incorporación de capacidades avanzadas de razonamiento, acceso a datos corporativos y cumplimiento normativo **sin requerir infraestructura propia de entrenamiento**. A continuación, se presentan algunos ejemplos representativos:

- ▶ **Microsoft 365 Copilot – Agentes y extensibilidad.** Permite crear agentes especializados dentro del ecosistema Microsoft, que actúan sobre datos empresariales en Microsoft Graph, Word, Excel, Teams o conectores externos. Soporta dos enfoques: agentes declarativos (configuración de instrucciones, datos, acciones) y agentes de motor personalizado (hosting externo, modelos propios o especializados) [52]. Algunas de las ventajas residen en la reducción de complejidad, aumentar velocidad de mercado, mejora de la confiabilidad y el cumplimiento, etc. Algunos ejemplos que se pueden crear son agentes de finanzas, gestión de facturas, automatización de procesos internos, etc. [53].
- ▶ **Expert.ai – Solución de IA vertical para dominios específicos.** Propuesta de un mayor control, explicabilidad y precisión semántica para sectores regulados o de alto riesgo (legal, compliance) [54].
- ▶ **Servicios:** existen algunas plataformas como Zfort Group [55] o Scopic [56] que ofrecen desarrollos a medida o de integración para sectores como salud, finanzas, legal, etc.
- ▶ **Creación de agentes multimodales para dominios complejos.** Este tipo de soluciones ilustra hacia dónde va el mercado: agentes “expertos de dominio” que combinan modelos base + contexto + herramientas especializadas.

Otras tendencias: Model Routers y SLM

Dentro del ámbito de la **optimización de costes y recursos** al realizar tareas con LLM, destaca la emergencia de los **model routers**, una categoría de herramientas y arquitecturas diseñadas para **asignar dinámicamente** las peticiones al modelo más adecuado según el tipo de tarea, la complejidad del input y el coste por token. La mayoría de estas soluciones pueden entenderse como una extrapolación del paradigma **Mixture of Experts (MoE)** [60] al ámbito de los LLM. En este enfoque, un **conjunto de modelos expertos**, cada uno optimizado para tareas o dominios **específicos y de tamaño reducido**, genera predicciones que son posteriormente ponderadas por un **modelo de enrutamiento** (*gating network model*), el cual determina qué peso debe tener cada experto en la respuesta final.

Otra tendencia destacable es el **creciente interés en los SLM** (*Small Language Models*) dentro del ecosistema IA [64]. Los SLM son **modelos más ligeros** que los grandes LLM generalistas. A diferencia de estos, los SLM están pensados para **casos de uso específico** en los que la **privacidad, latencia y eficiencia** son prioritarias [56]. Es posible realizar *fine-tuning* a una **fracción del coste** de un LLM usando **hardware accesible** (GPUs de 24-48 GB), con los despliegues integrados en pipelines RAG o agentes internos.

Retos empresariales

El desarrollo y la integración de modelos de lenguaje especializados ofrecen un enorme potencial de valor, pero también plantean una serie de retos estratégicos, operativos y éticos que se deben afrontar para lograr una adopción efectiva y sostenible.

Disponibilidad y calidad de los datos de dominio

Los LLM especializados dependen de datos específicos y de alta calidad. Sin embargo, muchas organizaciones enfrentan barreras para recopilar, limpiar o estructurar la información necesaria. Algunos retos claves son equilibrar la necesidad de datos relevantes con la protección de información sensible y la confidencialidad corporativa.

Gobernanza, cumplimiento y seguridad de la información

El uso de modelos que manejan conocimiento interno o información regulada implica altos estándares de gobernanza. Aparecen los siguientes retos clave: asegurar trazabilidad, control de acceso, y cumplimiento con normativas como GDPR o ISO 27001, especialmente cuando los modelos interactúan con datos personales o confidenciales.

Mantenimiento y actualización del conocimiento especializado

Un modelo de dominio puede volverse obsoleto rápidamente si no se actualiza con nuevos marcos normativos, criterios técnicos o cambios en la industria. Algunos de los retos clave son establecer procesos continuos de actualización del modelo y de la base de conocimiento asociada.

Integración con sistemas y flujos de trabajo existentes

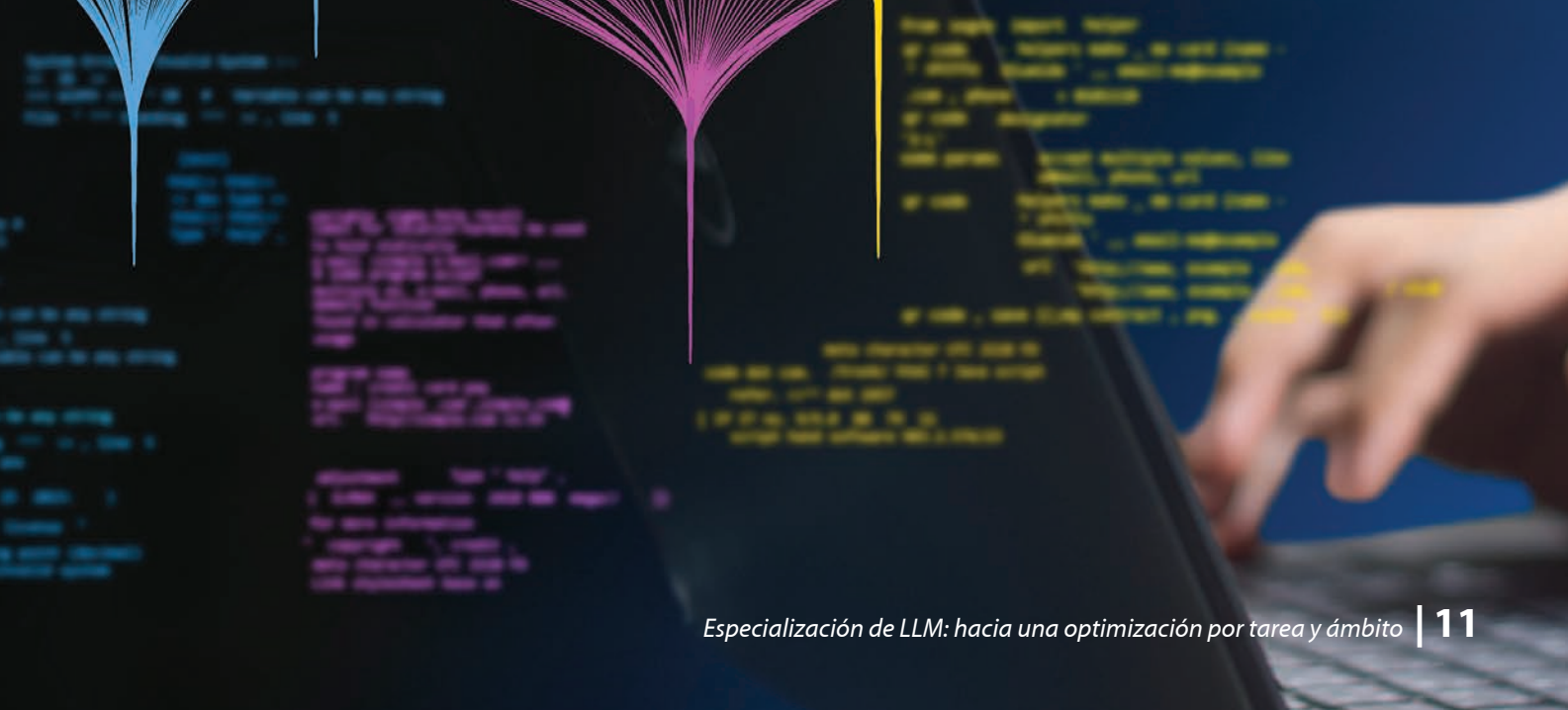
Adoptar un LLM especializado no se trata solo del modelo, sino de su integración dentro del ecosistema digital de la organización (ERP, CRM, intranet, sistemas documentales, etc.). Como principal reto clave aparece definir una arquitectura que permita integrar el modelo sin generar fricciones operativas ni duplicidades.

Escalabilidad y coste operativo

La personalización, el *fine-tuning* y el mantenimiento de modelos pueden requerir una infraestructura costosa o dependiente de terceros. Como principales retos clave surgen evaluar el retorno de inversión y definir una estrategia de escalado eficiente —por ejemplo, combinar LLM especializados con agentes modulares o con soluciones *off-the-shelf*.

Equilibrio entre personalización y dependencia de proveedores

El mercado actual ofrece numerosas plataformas propietarias (Copilot, Vertex AI, Claude, etc.) junto con opciones abiertas (Llama, Mistral, etc.). Algunos retos clave son encontrar el equilibrio adecuado entre aprovechar capacidades comerciales maduras y mantener soberanía sobre los datos y modelos.



Conclusiones

La **especialización** representa una de las etapas naturales en la maduración de la inteligencia artificial dentro de los **entornos productivos**. Los modelos de lenguaje generalistas continúan siendo herramientas extraordinariamente versátiles, idóneas para la experimentación rápida y la cobertura de una amplia gama de tareas, pero su **rendimiento** y su **escalabilidad** tienden a disminuir cuando se requiere un alto grado de **precisión, trazabilidad y comprensión contextual del dominio**.

En este escenario, estrategias de **context engineering** (como ICL y, especialmente, RAG) y las técnicas de **ajuste eficiente de parámetros** (PEFT mediante LoRA, QLoRA o adapters) son las más adoptadas. La elección óptima entre estas opciones no reside en priorizar una sobre otra, sino en diseñar **arquitecturas híbridas** donde cada técnica aporte su valor específico: RAG para incorporar conocimiento actualizado con transparencia y verificabilidad; PEFT para consolidar razonamiento especializado, alineación normativa y estilo; e ICL para facilitar la experimentación ágil o ajustar comportamientos concretos sin requerir despliegues complejos.

Además, la **estandarización del tool use**, impulsada por iniciativas como el *Model Context Protocol* (MCP), constituye un paso decisivo hacia la ampliación de las capacidades de los LLM, al permitirles interactuar con fuentes externas y acceder de forma controlada a información en tiempo real. Esta integración funcional no solo incrementa su utilidad práctica, sino que sienta las bases para el desarrollo de **Multi Agent Systems (MAS)** que pueden mejorar procesos como el mismo RAG.

Es destacable a su vez la evolución de conceptos como los **Small Language Models (SLM)** y los **model routers**, que introducen un enfoque **right-sizing** que permite que el paso a producción de sistemas de IA sea más eficiente.

También cabe destacar el crecimiento de **plataformas integradas**, el cual reduce las barreras técnicas de adopción, democratizando el uso de LLM especializados, aunque plantea nuevos desafíos en materia de gobernanza, transparencia y soberanía de datos. La verdadera ventaja competitiva no residirá únicamente en el acceso al modelo, sino en la capacidad de orquestar de manera estratégica su **integración, especialización y supervisión continua**.

Como última síntesis, la especialización de los modelos de lenguaje marca el tránsito desde la exploración hacia la consolidación de la inteligencia artificial como infraestructura crítica en los entornos productivos. El futuro de los LLM no dependerá únicamente de su **tamaño o potencia**, sino de su capacidad para **integrarse** de forma inteligente con **datos, herramientas y contextos específicos**. La combinación estratégica de enfoques como RAG, PEFT e ICL, junto con la estandarización del uso de herramientas y el auge de arquitecturas híbridas y escalables, permitirá diseñar sistemas más eficientes, auditables y adaptados a las necesidades reales de cada dominio. En última instancia, la competitividad se definirá por la madurez con que las organizaciones gestionen este ecosistema: cómo gobiernan sus modelos, cómo garantizan su trazabilidad y cómo alinean su evolución técnica con principios éticos y de soberanía digital.



Autores

Ernestina Menasalvas (UPM)
Manuel Ángel Guzmán (Management Solutions)
Rodrigo Lojero (Management Solutions)

Bibliografía

- [1] E. Mollick, «Scaling: The State of Play in AI,» One Useful Thing, September 2024. [En línea]. Available: <https://www.oneusefulthing.org/p/scaling-the-state-of-play-in-ai>.
- [2] A. McConnon, «Are bigger language models always better?,» IBM, [En línea]. Available: <https://www.ibm.com/think/insights/are-bigger-language-models-better>.
- [3] «ChatGPT vs Claude vs Gemini: ¿Cuál elegir en 2025 para proyectos de IA?,» DatiLab, September 2025. [En línea]. Available: <https://datilab.com/blog/chatgpt-vs-claude-vs-gemini-cual-elegir-2025-proyectos-ia>.
- [4] L. Wei, Z. Ying, M. He, Y. Chen, Q. Yang, Y. Hong, J. Lu, K. Zheng, S. Zhang, X. Li, W. Huang y Y. Chen, «Diabetica: Adapting Large Language Model to Enhance Multiple Medical Tasks in Diabetes Care and Management,» SCI-FM, ICLR 2025, 10.48550/arXiv.2409.13191, 2025.
- [5] H. Bansal, A. Hosseini, R. Agarwal, V. Q. T. Kazemi y Mehran, «Smaller, Weaker, Yet Better: Training LLM Reasoners via Compute-Optimal Sampling,» Google DeepMind, 2024.
- [6] Y. Lu, B. Yao, S. Zhang, Y. Wang, P. Zhang, T. Lu, T. J.-J. Li y D. Wang, «Human Still Wins over LLM: An Empirical Study of Active Learning on Domain-Specific Annotation Tasks,» 2023.
- [7] D. Anh-Hoang, V. Tran y L.-M. Nguyen, «Survey and analysis of hallucinations in large language models: attribution to prompting strategies or model behavior,» Front. Artif. Intell., 2025.
- [8] D. R. Rivera, «Evitar las alucinaciones de la GenAI,» VinculoTic, August 2025. [En línea]. Available: <https://vinculotic.com/salud/evitar-las-alucinaciones/>.
- [9] X.-K. Wu, M. Chen, W. Li, R. Wang, L. Lu, J. Liu, K. Hwang, Y. Hao, Y. Pan, Q. Meng, K. Huang, L. Hu, M. Guizani, N. Chao, G. Fortino, F. Lin, Y. Tian y D. Niyato, «LLM Fine-Tuning: Concepts, Opportunities, and Challenges,» Big Data and Cognitive Computing., 2025.
- [10] Q. Dong, L. Li, D. Dai, C. Zheng, J. Ma, R. Li, H. Xia, J. Xu, Z. Wu, T. Liu, B. Chang, X. Sun, L. Li y Z. Sui, «A Survey on In-context Learning,» arXiv, 2023.
- [11] W. Fan, Y. Ding, L. Ning, S. Wang, H. Li, D. Yin, T.-S. Chua y Q. Li, «A Survey on RAG Meeting LLM: Towards Retrieval-Augmented Large Language Models,» arXiv, 2024.
- [12] H. Chang, J. Park, S. Ye, S. Yang, Y. Seo, D.-S. Chang y M. Seo, «How Do Large Language Models Acquire Factual Knowledge During Pretraining?,» arXiv, 2024.
- [13] K. Buchholz, «The Extreme Cost Of Training AI Models,» Forbes, August 2024. [En línea]. Available: <https://www.forbes.com/sites/katharinabuchholz/2024/08/23/the-extreme-cost-of-training-ai-models/>.
- [14] D. Reigns, «Machine Learning Model Training Cost Statistics,» About Chromebooks, September 2025. [En línea]. Available: <https://www.aboutchromebooks.com/machine-learning-model-training-cost-statistics/>.
- [15] N. J. Prottasha, U. R. Chowdhury, S. Mohanto, T. Nuzhat, A. A. Sami, M. S. Ali, M. S. I. Sobuj, H. Raman, M. Kowsher y O. O. Garibay, «PEFT A2Z: Parameter-Efficient Fine-Tuning Survey for Large Language and Vision Models,» arXiv:2504.14117, 2025.
- [16] Y. Zhai, S. Tong, X. Li, M. Cai, Q. Qu, Y. J. Lee y Y. Ma, «Investigating the Catastrophic Forgetting in Multimodal Large Language Model Fine-Tuning,» Proceedings of Machine Learning Research, 2024.
- [17] Y. Luo, Z. Yang, F. Meng, Y. Li y J. Z. a. Y. Zhang, «An Empirical Study of Catastrophic Forgetting in Large Language Models During Continual Fine-Tuning,» IEEE Transactions on Audio, Speech and Language Processing, vol. 33, pp. 3776-3786, 2025.
- [18] Z. Han, C. Gao, J. Liu, J. Zhang y S. Q. Zhang, «Parameter-Efficient Fine-Tuning for Large Models: A Comprehensive Survey,» arXiv:2403.14608, 2024.
- [19] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang y W. Chen, «LoRA: Low-Rank Adaptation of Large Language Models,» arXiv:2106.09685, 2021.
- [20] T. Dettmers, A. Pagnoni, A. Holtzman y L. Zettlemoyer, «QLoRA: Efficient Finetuning of Quantized LLM,» NeurIPS Proceedings, 2023.
- [21] Z. Hu, L. Wang, Y. Lan, W. Xu, E.-P. Lim, L. Bing, X. Xu, S. Poria y R. K.-W. Lee, «LLM-Adapters: An Adapter Family for Parameter-Efficient Fine-Tuning of Large Language Models,» EMNLP , 2023.
- [22] K. Chen, Y. Pang y Z. Yang, «Parameter-Efficient Fine-Tuning With Adapters,» arXiv:2405.05493, 2024.
- [23] «Low-rank adaptation (LoRA) fine tuning,» IBM, October 2025. [En línea]. Available: <https://dataplatfom.cloud.ibm.com/docs/content/wsj/analyze-data/fm-tuning-methods-lora.html?context=wx&audience=wdp>.
- [24] «Recomendaciones de LoRA y QLoRA para LLM,» Google Cloud, October 2025. [En línea]. Available: <https://docs.cloud.google.com/vertex-ai/generative-ai/docs/model-garden/lora-qlora?hl=es-419>.
- [25] «¿Qué es la adaptación de bajo rango (LoRA)?,» Cloudflare, [En línea]. Available: [https://www.cloudflare.com/es-es/learning/ai/what-is-lora/#:~:text=La%20adaptaci%C3%B3n%20de%20bajo%20rango%20\(LoRA\)%20es%20un%20m%C3%A9todo%20para%20personalizar%20modelos%20para%20contextos%20espec%C3%ADficos..](https://www.cloudflare.com/es-es/learning/ai/what-is-lora/#:~:text=La%20adaptaci%C3%B3n%20de%20bajo%20rango%20(LoRA)%20es%20un%20m%C3%A9todo%20para%20personalizar%20modelos%20para%20contextos%20espec%C3%ADficos..)
- [26] L. Mei, J. Yao, Y. Ge, Y. Wang, B. Bi, Y. Cai, J. Liu, M. Li, Z.-Z. Li, D. Zhang, C. Zhou, J. Mao, T. Xia, J. Guo y S. Liu, «A Survey of Context Engineering for Large Language Models,» arXiv:2507.13334, 2025.
- [27] E. Kjosbakken, «How To Significantly Enhance LLM by Leveraging Context Engineering,» Towards Data Science, July 2025. [En línea]. Available: <https://towardsdatascience.com/how-to-significantly-enhance-llm-by-leveraging-context-engineering-2/>.
- [28] A. Zeichick, «¿Qué es la generación aumentada de recuperación (RAG)?,» OCl, September 2023. [En línea]. Available: <https://www.oracle.com/es/artificial-intelligence/generative-ai/retrieval-augmented-generation-rag/>.
- [29] J. Varughese, «¿Qué es el aprendizaje en contexto?,» IBM, [En línea]. Available: <https://www.ibm.com/es-es/think/topics/in-context-learning>. [Último acceso: November 2025].
- [30] X. Amatriain, «Prompt Design and Engineering: Introduction and Advanced Methods,» arXiv:2401.14423, 2024.
- [31] T. Feng, Y. Shen y J. You, «GraphRouter: A Graph-based Router for LLM Selections,» arXiv:2410.03834, 2024.
- [32] H. Han, Y. Wang, H. Shomer, K. Guo, J. Ding, Y. Lei, M. Halappanavar, R. A. Rossi, S. Mukherjee, X. Tang, Q. He, Z. Hua, B. Long, T. Zhao, N. Shah, A. Javari, Y. Xia y J. Tang, «Retrieval-Augmented Generation with Graphs (GraphRAG),» arXiv:2501.00309, 2024.
- [33] Y. Zhuang, C. Singh, L. Liu, J. Shang y J. Gao, «Vector-ICL: In-context Learning with Continuous Vector Representations,» arXiv:2410.05629, 2024.

- [34] C. Highmore, «In-Context Learning in Large Language Models: A Comprehensive Survey.» 10.20944/preprints202407.0926.v1. , 2024.
- [35] Z. Li, Z. Xu, L. Han, Y. Gao, S. Wen, D. Liu, H. Wang y D. N. Metaxas, «Implicit In-context Learning.» arXiv:2405.14660, 2024.
- [36] B. Sarmah, D. Mehta, B. Hall, R. Rao, S. Patel y S. Pasquali, «HybridRAG: Integrating Knowledge Graphs and Vector Retrieval Augmented Generation for Efficient Information Extraction.» In Proceedings of the 5th ACM International Conference on AI in Finance (ICAIF '24), 2024.
- [37] X. Hou, Y. Zhao, S. Wang y H. Wang, «Model Context Protocol (MCP): Landscape, Security Threats, and Future Research Directions.» arXiv:2503.23278, 2025.
- R. MacManus, «No, MCP Hasn't Killed RAG — in Fact, They're Complementary.» The New Stack, May 2025. [En línea]. Available: <https://thenewstack.io/no-mcp-hasnt-killed-rag-in-fact-theyre-complementary/>.
- [38] C. Vidal, «RAFT: A new way to teach LLM to be better at RAG.» Azure AI Foundry Blog, March 2024. [En línea]. Available: <https://techcommunity.microsoft.com/blog/azure-ai-foundry-blog/raft-a-new-way-to-teach-llm-to-be-better-at-rag/4084674>.
- [39] S. Schürch, «How to Make Your LLM More Accurate with RAG & Fine-Tuning.» Towards Data Science, March 2025. [En línea]. Available: <https://towardsdatascience.com/how-to-make-your-llm-more-accurate-with-rag-fine-tuning/>.
- [40] «In-Context Learning: Extreme vs. Fine-Tuning, RAG.» Meta Quantum Today, May 2024. [En línea]. Available: <https://meta-quantum.today/?p=2990>.
- [41] O. Kamath, «Main Page.» MeetCody.AI, [En línea]. Available: <https://meetcody.ai/es/blog/rag-como-servicio-desbloquea-la-ia-generativa-para-tu-empresa/>. [Último acceso: November 2025].
- [42] «RAG vs Fine-Tuning for LLM: A Comprehensive Guide with Examples.» Hugging Face, August 2024. [En línea]. Available: <https://huggingface.co/blog/airabbitX/rag-vs-fine-tuning-for-llm-a-com>.
- [43] J. Ferrer, «Fine-Tuning LLM: A Guide With Examples.» Data Camp, December 2024. [En línea]. Available: <https://www.datacamp.com/tutorial/fine-tuning-large-language-models>.
- [44] «Model optimization.» OpenAI, [En línea]. Available: <https://platform.openai.com/docs/guides/model-optimization>. [Último acceso: November 2025].
- [45] Z. Gekhman, G. Yona, R. Aharoni, M. Eyal, A. Feder, R. Reichart y J. Herzig, «Does Fine-Tuning LLM on New Knowledge Encourage Hallucinations?» EMNLP 2024, 2024.
- [46] O. Ovadia, M. Brief, M. Mishaeli y O. Elisha, «Fine-Tuning or Retrieval? Comparing Knowledge Injection in LLM.» arXiv:2312.05934v3 , 2023.
- [47] H. Soudani, E. Kanoulas y F. Hasibi, «Fine Tuning vs. Retrieval Augmented Generation for Less Popular Knowledge.» 12-22. 10.1145/3673791.3698415.
- [48] «RAG vs. fine-tuning: Choosing the right method for your LLM.» Super Annotate, August 2024. [En línea]. Available: <https://www.superannotate.com/blog/rag-vs-fine-tuning#:~:text=RAG%20is%20a%20good%20fit,that%20require%20specialized%2C%20precise%20responses..>
- [49] I. Mienye, N. Jere, G. Obaido, O. Ogunraku, Esenogho y C. Modisane, «Large language models: an overview of foundational architectures, recent trends, and a new taxonomy.» Discover Applied Sciences. 7. 10.1007/s42452-025-07668-w, 2025.
- [50] B. Gao, X. Wang, Y. Yang y D. Clifton, «Optimization-Inspired Few-Shot Adaptation for Large Language Models.» 10.48550/arXiv.2505.19107, 2025.
- [51] «Agentes para Microsoft 365 Copilot.» Microsoft, [En línea]. Available: <https://www.microsoft.com/es-es/microsoft-365-copilot/agents>. [Último acceso: November 2025].
- [52] «Managed Agents in Copilot Studio: Everything You Need to Know.» Global Sharepoint, October 2025. [En línea]. Available: <https://global-sharepoint.com/copilot/document-processors-managed-agent/>.
- [53] Alantra, «From restructuring to scalable, profitable growth.» Alantra, May 2025. [En línea]. Available: https://media.expert.ai/expertai/uploads/2020/08/20250528_Alantra_ExpertAi_From-restructuring-to-scalable-profitable-growth-20250528.pdf?
- [54] «Main Page.» ZFORT Group, [En línea]. Available: <https://www.zfort.com/>. [Último acceso: November 2025].
- [55] «Maximize Automation & Enhance Customer Interactions with LLM.» Scopic, [En línea]. Available: <https://scopicsoftware.com/llm-development-services/>. [Último acceso: November 2025].
- [56] Y. Zhang, X. Zhao, Z. Wang, G. Cheng, Y. Xu, S. Deng y J. Yin, «LightRouter: Towards Efficient LLM Collaboration with Minimal Overhead.» arXiv:2505.16221v1, 2025.
- [57] D. Stripelis, Z. Hu, J. Zhang, Z. Xu, A. D. Shah, H. Jin, Y. Yao, S. Avestimehr y C. He, «TensorOpera Router: A Multi-Model Router for Efficient LLM Inference.» EMNLP 2024, 2024.
- [58] T. Feng, Y. Shen y J. You, «GraphRouter: A Graph-based Router for LLM Selections.» 10.48550/arXiv.2410.03834. , 2024.
- [59] S. Masoudnia y R. Ebrahimpour, «Mixture of experts: a literature survey.» Artificial Intelligence Review. 42. 10.1007/s10462-012-9338-y. , 2014.
- [60] «Model router for Azure AI Foundry (preview).» Microsoft, 2025. [En línea]. Available: <https://learn.microsoft.com/en-us/azure/ai-foundry/openai/concepts/model-router>. [Último acceso: November 2025].
- [61] «Main page.» Storytell.ai, [En línea]. Available: <https://storytell.ai/>. [Último acceso: November 2025].
- [62] «NVIDIA AI Blueprints.» Github, [En línea]. Available: <https://github.com/NVIDIA-AI-Blueprints/llm-router>. [Último acceso: November 2025].
- [63] S. Savvov, «Your Company Needs Small Language Models.» Towards Data Science, December 2024. [En línea]. Available: <https://towardsdatascience.com/your-company-needs-small-language-models-d0a223e0b6d9/>.
- [64] J. Ferrer, «Todo lo que sabemos sobre GPT-5.» Datacamp, February 2025. [En línea]. Available: <https://www.datacamp.com/es/blog/everything-we-know-about-gpt-5>.



POLITÉCNICA

UNIVERSIDAD
POLITÉCNICA
DE MADRID

MS^o Management
Solutions
Making things happen

La Universidad Politécnica de Madrid es una Entidad de Derecho Público de carácter multisectorial y pluridisciplinar, que desarrolla actividades de docencia, investigación y desarrollo científico y tecnológico.

www.upm.es

Management Solutions es una firma internacional de consultoría, centrada en el asesoramiento de negocio, finanzas, riesgos, organización, tecnología y procesos, que opera en más de 50 países y con un equipo de más de 4.000 profesionales que trabajan para más de 2.200 clientes en el mundo.

www.managementsolutions.com

Para más información visita

blogs.upm.es/catedra-idanae/