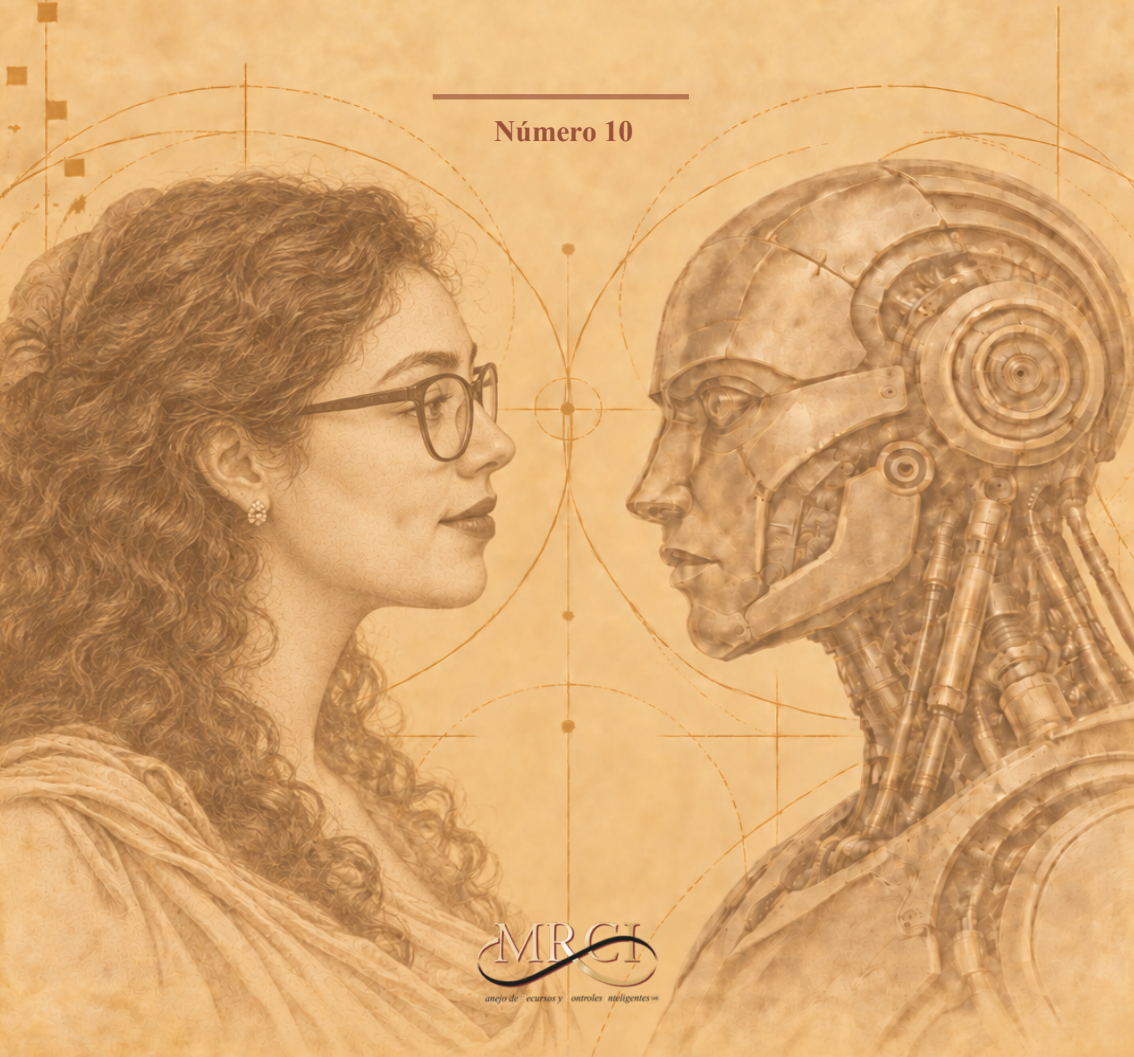
COLECCIÓN

MITOS Y REALIDADES DE LA IA

Entre la promesa y la realidad

CUANDO UN MODELO MÁS GRANDE MEJORA NUESTRO TRABAJO Y CUÁNDO NO

Número 10



MRCI

anejo de ecursos y ontroles ntéligentes

DR. JOSÉ ANTONIO PÉREZ RAMOS

CUANDO UN MODELO MÁS GRANDE MEJORA NUESTRO TRABAJO Y CUÁNDO NO

SOBRE EL AUTOR

Doctor en Ciencias de lo Fiscal por el Instituto de Especialización para Ejecutivos. Maestro en Fiscal por la Universidad Autónoma "Benito Juárez" de Oaxaca. Maestro en Estudios Avanzados en Derechos Humanos por la Global Open University. Licenciado en Contaduría Pública por la Universidad Autónoma "Benito Juárez" de Oaxaca y Licenciado en Filosofía por la Universidad Nacionalista de México.

Socio fundador y director general de la Firma Manejo de Recursos y Controles Inteligentes (Firma MRCI). Fiscalista del año 2009 por la revista *Defensa Fiscal*. Doctor honoris causa por la Asamblea General del Claustro Doctoral Iberoamericano (2016). Doctor honoris causa por el Instituto Mexicano de Líderes de Excelencia (2017). Doctor honoris causa por 1Million Startups, Latinomics, Leaderships Forum y la Fundación Humanist World (2018). Doctor honoris causa por el Claustro de Doctores de la AISOL (American International School of Law) (2019).

Autor de diversas obras, entre las que destacan *Ius mutantur o del derecho que cambia* (Tirant Lo Blanch, 2025), *Futurología* (MRCI, 2025) y *Hasta que la muerte nos separe* (MRCI, 2025). Ha participado como coordinador en *Cuestiones tributarias. Problemas y controversias en el México actual* (Tirant Lo Blanch, 2023), *El costo de la justicia* (Apex Iuris, 2019), *Remuneraciones estratégicas inteligentes* (MRCI, 2015) y *Retos de la justicia tributaria* (Tirant Lo Blanch-AMDF, 2025).

Titular de *Pie de Página*, programa para la divulgación de los derechos humanos en México, a través de la plataforma Legal Cloud. Expositor en foros nacionales e internacionales. Colaborador en diversas revistas jurídicas, entre las que destacan *Defensa Fiscal* y *Legal Cloud*.

CUANDO UN MODELO MÁS GRANDE MEJORA NUESTRO TRABAJO Y CUÁNDO NO

DR. JOSÉ ANTONIO PÉREZ RAMOS

PRIMERA EDICIÓN, ATEMPORAL

Derechos reservados, propiedad de
Dr. José Antonio Pérez Ramos

Comentarios y opiniones: investigacion@mrci.com.mx

Título original: Cuando un modelo más grande mejora nuestro trabajo y cuándo no.

Colección mitos y realidades de la inteligencia artificial.

Entre la promesa y la realidad.

Autor: Dr. José Antonio Pérez Ramos.

Queda prohibida la reproducción total y parcial de esta obra denominada CUANDO UN MODELO MÁS GRANDE MEJORA NUESTRO TRABAJO Y CUÁNDO NO, por cualquier medio, sin autorización escrita del autor.

PRINTED IN MEXICO
IMPRESO EN MÉXICO

ÍNDICE

LA PREGUNTA QUE PRECEDE A LA COMPRA..... - 7 -

SIETE MAGNITUDES QUE NO SON INTERCAMBIABLES ..... - 9 -

QUÉ ENSEÑAN LAS INVESTIGACIONES DE ESCALAMIENTO .....- 14 -

ÓPTIMO PARA ENTRENAR Y ÓPTIMO PARA UTILIZAR.....- 18 -

TRES MANERAS DIFERENTES DE SER MÁS LIGERO .....- 21 -

DAR MÁS RECURSOS A UNA RESPUESTA.....- 26 -

CASO DE UNA REVISIÓN QUE CAMBIA EL COSTE DEL LOTE.....- 29 -

CASO DE DISTRIBUCIÓN ENTRE MODELOS .....- 33 -

CÓMO SABER SI LA MEJORA VIENE DEL MODELO .....- 37 -

FALLOS QUE EL TAMAÑO NO CORRIGE POR SÍ SOLO .....- 41 -

UN PROCEDIMIENTO DE SELECCIÓN REVISABLE .....- 42 -

OBJECIONES QUE MERECE LA PENA CONSERVAR.....- 45 -

CONCLUSIÓN.....- 47 -

REFERENCIAS .....- 47 -

La pregunta que precede a la compra

Imaginemos una oficina que prepara informes a partir de expedientes y observa que algunas respuestas omiten excepciones importantes; alguien propone utilizar el modelo con mayor número de parámetros disponible, mientras otra persona considera suficiente modificar las instrucciones, aunque ninguna de las dos propuestas identifica todavía dónde se pierde la información ni qué cambio permitiría recuperar las excepciones.

La discusión adquiere contenido cuando se formula una pregunta comprobable: ¿un cambio de modelo reduce los errores relevantes en estos expedientes, bajo las mismas condiciones de acceso a documentos y con un tiempo de revisión aceptable?; sin esa delimitación, el tamaño funciona como sustituto de una evaluación que no se ha realizado.

Defendemos que la capacidad adicional debe justificarse por su contribución al resultado completo, desde la entrada de información hasta la entrega verificada; una mejora en la redacción inicial puede ser valiosa, pero deja de ser una ventaja suficiente si requiere más correcciones, introduce fallos graves o impide cumplir el plazo de trabajo.

La palabra «grande» necesita, además, un término de comparación: un modelo puede tener más parámetros totales y activar menos por token, ocupar menos memoria gracias a una representación numérica distinta o emplear más cómputo al responder sin cambiar sus pesos; confundir estas magnitudes impide explicar tanto los beneficios como los costes.

Método y alcance

Se realizó una revisión documental explicativa de ocho investigaciones primarias, consultando sus textos completos para cotejar los pasajes utilizados; se eligieron versiones identificables de trabajos publicados entre 2015 y 2024 que permiten examinar mecanismos y condiciones de comparación, sin presentar la selección como revisión sistemática exhaustiva ni como descripción de toda la frontera tecnológica.

La fecha de corte documental de esta versión es el 2 de octubre de 2026, hora de Oaxaca; las cifras históricas conservan el año, modelo y contexto de la investigación de origen, mientras los ejemplos numéricos propios se

identifican como hipotéticos y no representan pruebas realizadas con productos comerciales.

El análisis se concentra en modelos de lenguaje y aplicaciones documentales, porque permiten distinguir con claridad las decisiones que suelen confundirse al comparar tamaños; no generalizamos sus relaciones numéricas a sistemas de visión, robótica o cualquier otra familia de inteligencia artificial, ni establecemos precios vigentes, recomendaciones de compra de equipos o una clasificación universal de proveedores.

Dentro de la colección, este artículo estudia la elección de capacidad; la explicación detallada del entrenamiento y la inferencia corresponde al artículo 09, mientras la teoría general de evaluación y los límites de las clasificaciones se desarrollan en los artículos 61 y 62.

Siete magnitudes que no son intercambiables

Un parámetro es un valor numérico del modelo que interviene en sus operaciones y que puede haber sido ajustado durante el entrenamiento; contar parámetros describe una propiedad de la arquitectura entrenada, pero



no equivale a contar hechos verdaderos, documentos memorizados, conceptos comprendidos o tareas profesionales que el sistema puede completar correctamente.

En un modelo denso, grandes conjuntos de pesos participan de manera regular en el procesamiento de cada token; en una arquitectura dispersa, algunas partes se seleccionan condicionalmente, por lo que conviene distinguir el inventario total de parámetros de los que se activan para una unidad de procesamiento, con el alcance exacto que declare la publicación técnica.

El cómputo de entrenamiento representa las operaciones realizadas para ajustar el modelo y suele expresarse mediante FLOPs, es decir, operaciones de punto flotante; esta medida no es una factura, una duración ni una cantidad de energía, porque la conversión depende del equipo, su utilización y la implementación, además de las condiciones bajo las cuales se ejecutó el trabajo.

Los datos de entrenamiento constituyen otra dimensión: su cantidad puede expresarse en tokens, pero contar tokens no describe por completo su pertinencia, diversidad,



calidad ni repetición; tampoco es válido equiparar una cantidad de tokens con una cantidad fija de palabras o de documentos, ya que intervienen el texto y el método de representación utilizado.

El presupuesto de inferencia corresponde a los recursos permitidos al producir una respuesta: longitud de entrada y salida, número de intentos, búsquedas, verificaciones o revisiones; mantener el mismo modelo y ampliar ese presupuesto puede modificar el resultado, de modo que una comparación de productos no siempre constituye una comparación de tamaños.

El desempeño en una tarea describe lo que el sistema logra bajo un procedimiento de prueba; el coste total de operación incorpora los recursos que efectivamente exige sostener ese desempeño, incluidas actividades que ocurren fuera del modelo, como preparar los documentos, revisar las salidas y corregir los fallos que llegan al proceso. La Tabla 1 organiza estas diferencias como criterios que pueden consultarse con un proveedor o utilizarse en un expediente de evaluación; sus ejemplos son conceptos de análisis y no especificaciones de un servicio particular.

Tabla 1. *Magnitudes para comparar capacidad y utilidad*

Magnitud	Qué informa	Qué no permite concluir por sí sola
Parámetros totales	Tamaño del conjunto de pesos	Exactitud en nuestra tarea
Parámetros activos	Pesos utilizados en una unidad de procesamiento	Memoria total de despliegue
Cómputo de entrenamiento	Operaciones destinadas al ajuste	Precio final de cada respuesta
Datos de entrenamiento	Exposición al material de aprendizaje	Calidad o actualidad de toda respuesta
Presupuesto de inferencia	Recursos destinados a resolver una solicitud	Corrección del resultado seleccionado
Desempeño por tarea	Resultado bajo condiciones documentadas	Superioridad en cualquier contexto
Coste de operación	Recursos del flujo dentro del límite contable	Cumplimiento de requisitos de calidad

Nota. Elaboración propia; las distinciones se aplican al análisis del artículo y requieren definiciones específicas al comparar sistemas reales.

El tamaño anunciado también tiene una definición

Antes de comparar dos cifras hay que comprobar si ambas incluyen los mismos componentes; Kaplan et al. (2020, sección 1.3) definen su variable de tamaño excluyendo las representaciones del vocabulario y de posición, por lo que trasladar sin aclaración esa cifra a otra publicación con un criterio distinto puede crear una comparación aparente.

Algo semejante ocurre con las denominaciones de familias comerciales o técnicas, que pueden condensar decisiones de diseño sin expresar una suma literal de componentes; una etiqueta es un identificador útil, pero el análisis requiere la ficha de arquitectura y la explicación del conteo, cuando esa información está disponible.

Si el proveedor no comunica el número de parámetros, la incertidumbre debe registrarse, sin deducir una cifra a partir de la velocidad o del precio; todavía es posible evaluar el servicio como sistema observable, aunque no

será posible atribuir de forma demostrada sus diferencias al tamaño interno.

Tampoco conviene confundir la longitud de contexto con la cantidad de parámetros: permitir una entrada más extensa cambia el material que una aplicación puede presentar al modelo, mientras la capacidad de usar correctamente ese material constituye una pregunta empírica adicional que debe probarse con los documentos y relaciones relevantes.

Qué enseñan las investigaciones de escalamiento

Kaplan et al. (2020, secciones 1.1, 1.2 y 2) estudian relaciones empíricas entre pérdida de predicción, parámetros, datos y cómputo en modelos neuronales de lenguaje; encuentran regularidades de tipo ley de potencia dentro de los regímenes examinados y proponen cómo distribuir recursos cuando el objetivo es mejorar esa pérdida bajo restricciones determinadas.

La pérdida resume el desacuerdo entre las probabilidades asignadas por el modelo y los tokens observados en una evaluación; reducirla indica una mejora según ese objetivo



estadístico, pero no equivale directamente a reducir en la misma proporción las omisiones de un informe, los errores de una clasificación o las horas de revisión de una oficina.

El interés de esa investigación no reside en prometer que cualquier crecimiento funcionará, sino en mostrar que ciertas mejoras pueden estudiarse mediante relaciones cuantitativas; sus fórmulas dependen del régimen experimental y de las restricciones consideradas, por lo que utilizarlas fuera de ese alcance exige una validación adicional.

Cuando más datos cambian la comparación

Hoffmann et al. (2022, secciones 3 y 4) revisan la distribución del cómputo entre tamaño y tokens de entrenamiento; sus experimentos llevan a recomendar un crecimiento aproximadamente equilibrado de ambas dimensiones para el objetivo de entrenamiento considerado, en contraste con asignaciones que favorecían mucho más el aumento de parámetros.

Como comprobación, entrenan Chinchilla con 70 mil millones de parámetros y más tokens, utilizando un

presupuesto de cómputo comparable al de Gopher, de 280 mil millones; el modelo menor obtiene mejores resultados en las evaluaciones comunicadas, lo que refuta la inferencia de que el mayor conteo basta para anticipar la superioridad de esos dos sistemas bajo aquel diseño experimental (Hoffmann et al., 2022, sección 4).

Este resultado no demuestra que reducir arbitrariamente los parámetros produzca una mejora; cambia conjuntamente la distribución de recursos, la exposición a datos y otros detalles de entrenamiento, como el optimizador y la tokenización, de modo que la conclusión pertinente es que el entrenamiento puede hacer mejor uso del presupuesto y que el número aislado no captura esa diferencia (Hoffmann et al., 2022, sección 4.1).

También sería incorrecto convertir una proporción empírica de entrenamiento en una receta universal para todos los modelos futuros; los objetivos, las arquitecturas, la disponibilidad de material y las condiciones de uso pueden cambiar, mientras la recomendación original responde a un problema concreto de optimización.



Una regularidad científica no es una garantía de contratación

Para una organización que adquiere un servicio ya entrenado, las leyes de escalamiento ayudan a formular hipótesis sobre capacidad y recursos, pero no sustituyen la comprobación del trabajo; la decisión relevante puede depender de una excepción que casi no modifica la pérdida promedio y, sin embargo, determina si un expediente queda utilizable.

Pensemos en dos generadores que redactan de manera semejante la mayor parte de un informe, aunque uno de ellos identifica con mayor frecuencia una incompatibilidad entre fechas; si esa incompatibilidad cambia la conclusión, una mejora pequeña en una métrica global podría representar una diferencia importante para la tarea, sin que exista una conversión general entre ambas medidas.

La relación también puede invertirse: un sistema puede mejorar su predicción del lenguaje y producir textos más fluidos sin resolver la operación que bloquea el proceso; la evaluación local necesita identificar qué cambió en los resultados, en vez de suponer que cualquier progreso



agregado debe trasladarse de forma automática al uso profesional.

Óptimo para entrenar y óptimo para utilizar

Un presupuesto de entrenamiento se consume antes del despliegue, mientras la inferencia puede repetirse durante la vida útil del modelo; por esa razón, minimizar los recursos necesarios para entrenar hasta cierto nivel no es necesariamente lo mismo que minimizar los recursos necesarios para entrenar y atender todas las solicitudes posteriores.

Sardana et al. (2024, secciones 2 y 6) incorporan la demanda de inferencia al problema de selección de tamaño y datos; su análisis favorece, en determinados regímenes de uso intenso, modelos menores entrenados durante más tiempo, y distingue además la optimización por operaciones de la que utiliza supuestos de coste monetario.

Los autores explicitan que su formulación requiere estimar la demanda futura y emplea la pérdida como aproximación de calidad; también examinan los límites de extrapolar

relaciones ajustadas con proporciones habituales de datos hacia regímenes de entrenamiento mucho más prolongado, de manera que su resultado no elimina la necesidad de comprobar las condiciones de aplicación (Sardana et al., 2024, secciones 2 y 5).

Para interpretar esa lógica no necesitamos reproducir una contabilidad de un laboratorio: una inversión inicial mayor puede resultar razonable si reduce suficientemente el gasto de cada uso posterior, pero esa ventaja depende del número de usos que realmente se produzcan y de que se mantenga el nivel de resultado requerido.

Una organización usuaria no siempre paga directamente el entrenamiento y puede contratar una tarifa por consumo; en tal situación, no debe sumar a su factura una estimación del gasto histórico del proveedor, porque mezclaría perspectivas contables y posiblemente contaría dos veces un coste ya incorporado en el precio del servicio.

Si la organización entrena o adapta un sistema propio, sí corresponde incluir los desembolsos y horas que deba asumir dentro del horizonte de la decisión; la pregunta contable es quién incurre en el recurso y para qué



alternativa, no si ese recurso aparece alguna vez en la cadena tecnológica completa.

Costes fijos y volumen real

Supongamos un ejemplo enteramente hipotético: preparar una solución pequeña especializada exige 1200 unidades monetarias adicionales, pero permite ahorrar 0.04 unidades por documento frente a la alternativa de referencia, manteniendo el mismo resultado aceptable; el punto de recuperación simple es 1200 dividido entre 0.04, equivalente a 30 000 documentos.

Con 10 000 documentos, el ahorro variable sería de 400 unidades y todavía no compensaría aquella preparación; con 50 000, ascendería a 2000 y dejaría 800 unidades de diferencia favorable, siempre que no aparezcan otros costes diferenciales, que no cambie la calidad y que el volumen previsto llegue a materializarse.

Este cálculo no incorpora financiación, impuestos ni el valor temporal del dinero, porque sirve para explicar una relación aritmética y no para valorar una inversión real; tampoco convierte 30 000 documentos en un umbral

general de adopción, pues depende por completo de los dos supuestos elegidos para construirlo.

La posibilidad de que un servicio cambie de versión antes de recuperar la inversión merece atención: una especialización muy costosa puede perder parte de su utilidad si los documentos o las reglas cambian, por lo que la duración del beneficio debe examinarse junto con la cantidad de solicitudes.

Tres maneras diferentes de ser más ligero

Reducir la carga de despliegue no significa siempre reducir el número de parámetros; es posible entrenar un modelo distinto para reproducir ciertos comportamientos, representar sus pesos con menos bits o activar sólo partes de una arquitectura, y cada ruta modifica componentes diferentes del problema.

Destilar una capacidad delimitada

Hinton et al. (2015, secciones 1 y 2) desarrollan un procedimiento de destilación en el que un modelo aprende de distribuciones de salida producidas por otro modelo o

un conjunto de modelos; esas distribuciones aportan información diferente de una etiqueta rígida, y los autores estudian la transferencia en reconocimiento de dígitos y en modelos acústicos.

El trabajo distingue el sistema más costoso que aporta las señales de aprendizaje del sistema que se pretende desplegar con mayor facilidad; constituye evidencia de que ciertos comportamientos pueden transferirse, pero no una demostración de que cualquier modelo pequeño conservará todas las capacidades de cualquier maestro ni de que los resultados de 2015 describan todas las técnicas actuales (Hinton et al., 2015, secciones 3 y 4).

La pregunta práctica es qué capacidad necesita preservarse y cómo se comprobará; un clasificador especializado en categorías estables puede ser una opción pertinente aunque no reproduzca la amplitud de un asistente general, siempre que las excepciones que importan para su cometido formen parte de la evaluación.

El modelo maestro tampoco debe convertirse en árbitro infalible: si la única definición de éxito consiste en imitarlo, algunos errores compartidos podrían contarse como



coincidencias satisfactorias; para el uso propuesto conviene contrastar el resultado con criterios externos a la pareja de modelos y mantener casos que no hayan guiado la preparación.

Cuantizar los pesos

Frantar et al. (2023, secciones 3 a 5) presentan GPTQ como método de cuantización posterior al entrenamiento y examinan modelos cuyos pesos se representan con menos bits; sus experimentos documentan intercambios entre precisión numérica, calidad medida y ejecución, por lo que respaldan la posibilidad de reducir almacenamiento en determinadas condiciones y no una promesa universal de conservación exacta.

Cuantizar cambia la representación numérica y puede conservar el mismo conteo de parámetros; tampoco basta con que un archivo sea menor para garantizar la misma reducción del tiempo de respuesta, porque la implementación y el equipo deben aprovechar la representación y porque otras partes de la ejecución pueden limitar el rendimiento (Frantar et al., 2023, secciones 1 y 5).



La Tabla 2 ilustra únicamente la memoria teórica para almacenar pesos de modelos hipotéticos, sin escalas auxiliares, estados temporales, activaciones, memoria de contexto ni sobrecarga del programa; sus valores se obtienen multiplicando parámetros por bits y dividiendo entre ocho para pasar a bytes.

Tabla 2. *Memoria idealizada de pesos según su representación*

Modelo hipotético	Representación de cada peso	Memoria de pesos en GB decimales
8000 millones de parámetros	16 bits	16
8000 millones de parámetros	8 bits	8
8000 millones de parámetros	4 bits	4
70 000 millones de parámetros	4 bits	35

Nota. Elaboración propia; 1 GB decimal equivale a 1000 millones de bytes y las cantidades no representan requisitos completos de memoria para ejecutar un modelo.

El primer cálculo es 8000 millones por 16 bits, dividido entre ocho, lo que produce 16 000 millones de bytes; pasar a cuatro bits reduce esa parte idealizada a una cuarta parte, aunque la memoria completa no tiene por qué disminuir en esa misma proporción.

El último renglón permite advertir otra diferencia: un modelo de 70 000 millones de parámetros a cuatro bits todavía requiere 35 GB decimales sólo para sus pesos ideales; utilizar una representación más compacta no elimina la necesidad de comprobar cuánto ocupa el sistema completo en el entorno de ejecución previsto.

Activar una parte del conjunto

Jiang et al. (2024, secciones 2 y 3) describen Mixtral como una mezcla dispersa de expertos en la que un mecanismo de selección elige dos bloques expertos por token y capa; el informe distingue aproximadamente 47 mil millones de parámetros totales y 13 mil millones activos, lo que



muestra por qué comparar sólo un conteo puede ocultar diferencias de arquitectura.

En este contexto, «experto» designa un bloque de la red y no a una persona especializada ni a un agente independiente que delibera; los autores también explican que distribuir esos bloques entre dispositivos introduce problemas de equilibrio de carga y comunicación, de modo que la reducción de actividad por token no suprime los requisitos del despliegue (Jiang et al., 2024, sección 2.1).

Por ello, una evaluación necesita pedir por separado memoria, cómputo y velocidad bajo la carga esperada; deducir que sólo deben almacenarse los parámetros activos confundiría la selección momentánea con el conjunto de pesos que el sistema podría necesitar en los siguientes pasos o solicitudes.

Dar más recursos a una respuesta

El tamaño permanece fijo durante una inferencia ordinaria, pero la aplicación puede permitir varios intentos, revisar una salida o usar un verificador para seleccionar entre candidatos; estas estrategias consumen recursos



adicionales y deben compararse como configuraciones completas, incluyendo la selección y sus posibles fallos.

Snell et al. (2024, secciones 3 a 7) estudian cómo asignar cómputo de inferencia en problemas matemáticos mediante revisiones y búsqueda con verificadores; la eficacia depende de la dificultad de la pregunta para el modelo, y sus resultados muestran condiciones en las que un modelo menor con más cómputo puede superar a uno mayor al igualar estimaciones de operaciones de entrenamiento e inferencia.

El estudio no demuestra que cualquier problema se resuelva repitiendo solicitudes: utiliza un dominio de evaluación concreto y reconoce que estimar la dificultad tiene un coste que sus experimentos no contabilizan plenamente; también encuentra límites importantes en las preguntas más difíciles (Snell et al., 2024, secciones 3.2, 4 y 7).

La consecuencia práctica es evaluar el procedimiento que se utilizará, no una posibilidad ideal de escoger retrospectivamente la mejor respuesta; si una prueba cuenta como éxito que exista una solución correcta entre



cien candidatos, todavía falta demostrar cómo un usuario o verificador la distinguirá sin conocer de antemano la respuesta.

La trampa de sumar oportunidades

Consideremos un ejercicio probabilístico hipotético con una probabilidad de éxito de 0.60 por intento y cuatro intentos independientes con idéntica probabilidad; la probabilidad de que exista al menos una respuesta correcta es 1 menos 0.40 elevado a cuatro, equivalente a 0.9744, es decir, 97.44 %.

Ese porcentaje no representa la probabilidad de que la respuesta elegida sea correcta; expresa la existencia de al menos un acierto bajo supuestos fuertes, y todavía requiere un mecanismo de selección que puede equivocarse, además de que los intentos reales pueden compartir errores y no satisfacer la independencia postulada.

Si los cuatro intentos cuestan 0.20 unidades cada uno y la selección añade 0.10, el procedimiento cuesta 0.90 unidades; una alternativa de una sola respuesta a 0.60



sería menos costosa por ejecución, aunque decidir entre ambas exige conocer la calidad final de sus resultados seleccionados y no sólo el número de oportunidades disponibles.

Solicitar que el sistema «piense más» tampoco documenta por sí mismo qué recursos utilizó ni qué mejoró; una comparación reproducible necesita registrar el presupuesto configurado y el consumo observado cuando sea accesible, y declarar lo desconocido cuando la interfaz no permita inspeccionarlo.

Caso de una revisión que cambia el coste del lote

Consideremos un caso conceptual sencillo y desarrollemos sus condiciones: una organización debe procesar 1000 documentos, con una opción compacta C y otra de mayor capacidad G, ambas hipotéticas; el precio de ejecución del lote es de 100 unidades para C y 300 para G, pero esa información todavía no permite elegir.

Supongamos que, dentro de este escenario construido, revisar y corregir las salidas de C requiere 60 horas y las de G requieren 20, con una valoración de 10 unidades por

hora; los importes de revisión serían 600 y 200, respectivamente, y la suma de ejecución más revisión ascendería a 700 para C y 500 para G.

La diferencia favorable a G es de 200 unidades por lote, aun cuando su ejecución cuesta el triple; la explicación reside en las 40 horas de revisión evitadas, y no en una regla que asocie automáticamente mayor tamaño con menor necesidad de supervisión.

Tabla 3. *Coste hipotético de ejecución y revisión de 1000 documentos*

Partida	Opción C	Opción G
Ejecución en unidades monetarias	100	300
Horas de revisión y corrección	60	20
Revisión a 10 unidades por hora	600	200
Suma de ejecución y revisión	700	500

Nota. Elaboración propia; todos los datos son hipotéticos y la suma comprende sólo las partidas indicadas, sin equivaler automáticamente al coste total de propiedad.

El tiempo medio de revisión sería de 3.6 minutos por documento para C y 1.2 minutos para G, calculado al multiplicar las horas por 60 y dividir entre 1000; se trata de promedios del lote, por lo que podrían coexistir documentos de revisión muy breve con otros que concentren buena parte del trabajo.

Si cambia la valoración de la hora, también puede cambiar la conclusión económica: la ejecución adicional de G cuesta 200 unidades y ahorra 40 horas, de modo que ambas sumas se igualan a cinco unidades por hora; por encima de ese valor G cuesta menos en las dos partidas modeladas, y por debajo C cuesta menos.

La calidad debe examinarse antes de tratar ese umbral como criterio de selección; si una opción no cumple los requisitos mínimos después de la revisión disponible, no basta con que su suma resulte menor, y si el ahorro horario

se obtuvo omitiendo controles necesarios, no se ha demostrado una mejora comparable.

Lo que falta medir antes de adoptar

Una prueba real tendría que registrar la corrección inicial, el tiempo dedicado a detectar errores, las modificaciones y la calidad final; medir sólo cuánto tarda una persona en leer la respuesta puede ocultar trabajo que reaparece después, como resolver una contradicción, rehacer una tabla o atender una devolución del destinatario.

También conviene distinguir horas liberadas de dinero efectivamente ahorrado: una reducción del tiempo de revisión puede permitir asumir más trabajo o acortar plazos, pero no produce por sí sola una reducción equivalente de nómina; el beneficio dependerá de cómo se utilice la capacidad disponible y del objetivo de la organización.

Si G genera respuestas más largas, comparar tiempos sin controlar la extensión pertinente puede resultar engañoso; una salida extensa podría contener información útil o imponer lectura innecesaria, por lo que la evaluación debe

considerar la suficiencia del contenido y no premiar automáticamente la brevedad o la abundancia.

Por último, cada partida excluida debe quedar visible: integración, mantenimiento, preparación de documentos, incidencias y condiciones de contratación pueden modificar la decisión; la Tabla 3 demuestra que el precio de ejecución no basta, pero no pretende sustituir una contabilidad completa del servicio.

Caso de distribución entre modelos

Una alternativa a utilizar siempre el mismo modelo consiste en resolver una parte de las solicitudes con una opción compacta y remitir otras a una de mayor capacidad; esa distribución suele denominarse enrutamiento, aunque el nombre no demuestra que el sistema identifique bien los casos que necesitan ayuda.

Construyamos un segundo caso hipotético con 1000 solicitudes, de las cuales 800 son ordinarias y 200 difíciles; supongamos tasas de acierto de 98 % y 60 % para C en cada grupo, y de 99 % y 90 % para G, sin afirmar que estas tasas correspondan a ningún modelo real.



Si todas las solicitudes se atienden con C, el número esperado de aciertos es 800 por 0.98 más 200 por 0.60, equivalente a 904; con G, sería 800 por 0.99 más 200 por 0.90, equivalente a 972, por lo que las tasas esperadas serían 90.4 % y 97.2 %.

Ahora supongamos que C atiende inicialmente todo y un selector remite a G 150 solicitudes difíciles y 50 ordinarias, suponiendo que las tasas se mantienen en los subgrupos remitidos y retenidos; al conservar 750 ordinarias y 50 difíciles con C, y sustituir las restantes respuestas por las de G, el número esperado de aciertos es 735 más 30 más 49.5 más 135, equivalente a 949.5.

El valor fraccionario expresa una expectativa matemática, no medio documento observado; la tasa esperada del sistema combinado sería de 94.95 %, que mejora sobre C pero permanece por debajo de G, mientras la calidad del selector queda reflejada tanto en los casos difíciles que remite como en los que deja sin escalar.

Para calcular la ejecución, asignemos 0.10 unidades a cada uso de C, 0.30 a cada uso de G y 20 unidades al selector por lote; el sistema combinado cuesta 100 más 60



más 20, equivalente a 180, frente a 300 si se utiliza G directamente en todas las solicitudes.

El ahorro de 120 unidades no constituye todavía una decisión aceptable si el requisito mínimo propuesto es alcanzar 97 % de aciertos esperados en este escenario; la distribución de 200 solicitudes no cumple ese requisito, aunque sea más económica por ejecución.

Cuando escalar mucho deja de ahorrar

Incluso un selector ideal que reconociera las 200 solicitudes difíciles y las remitiera todas a G, dejando las 800 ordinarias con C, obtendría 784 más 180 aciertos esperados, equivalentes a 96.4 %; para llegar al 97 % tendría que mejorar también parte del grupo ordinario bajo las tasas supuestas.

Cada solicitud ordinaria trasladada de C a G añade 0.01 aciertos esperados, de modo que se necesitan 600 traslados adicionales para cubrir los seis aciertos faltantes; el total escalado sería de 800 solicitudes y la ejecución costaría 100 más 240 más 20, equivalente a 360 unidades.



En esta construcción, el sistema que comienza siempre con C termina costando más que G directo cuando se le exige alcanzar el umbral fijado; el resultado surge de los valores supuestos y del doble procesamiento, por lo que no demuestra que todo enrutamiento fracase, sino que su ahorro depende de la distribución de dificultades, el selector y el requisito de calidad.

Un selector que decide antes de ejecutar C tendría otra contabilidad y necesitaría otra prueba; del mismo modo, una regla de escalamiento que permita abstenerse o enviar documentos a una persona cambiaría las tasas, los recursos y el significado del resultado, por lo que no deben mezclarse esas arquitecturas dentro de una sola cifra comparativa.

Además, una tasa esperada de 97 % no garantiza que un lote real alcance ese porcentaje ni establece que sea suficiente para un trabajo de consecuencias importantes; el umbral utilizado es didáctico y la decisión real necesita considerar la incertidumbre de las mediciones y la gravedad de los fallos restantes.



Cómo saber si la mejora viene del modelo

Una comparación útil necesita mantener constantes los elementos que no desea estudiar, como documentos, permisos, instrucciones esenciales y definición de éxito; si una alternativa tiene acceso a la fuente completa y otra recibe fragmentos incompletos, sus resultados no permiten atribuir la diferencia únicamente a su tamaño.

Esto no obliga a imponer configuraciones artificialmente idénticas cuando cada sistema requiere una integración distinta; permite realizar dos preguntas complementarias, una sobre el efecto del modelo bajo condiciones controladas y otra sobre la mejor configuración completa que cada alternativa puede ofrecer con el presupuesto disponible.

Las dos preguntas deben conservarse separadas en el informe: una configuración optimizada puede ser la decisión operativa correcta aunque no identifique una causa aislada, mientras una prueba controlada puede explicar un mecanismo sin representar el servicio más conveniente para usar a diario.



Liang et al. (2022, secciones 1 y 4) proponen en HELM una evaluación mediante distintos escenarios y varias métricas, incluidas exactitud, calibración, robustez y eficiencia; el trabajo muestra la importancia de hacer visibles los intercambios entre criterios y de identificar lo que la evaluación no cubre, en lugar de reducir toda comparación a una posición única.

Tomamos esa orientación como apoyo para un procedimiento local de selección, no como validación de una plantilla universal; las pruebas de un despacho, una editorial o una oficina administrativa deben responder a sus propias entradas y consecuencias, sin afirmar que una clasificación pública haya medido necesariamente esas condiciones.

Un caso pareado de 200 expedientes

Supongamos una prueba hipotética en la que C y G reciben los mismos 200 expedientes y cada salida se clasifica con un criterio binario previamente definido; ambos aciertan en 170, sólo G acierta en 18, sólo C acierta en seis y ambos fallan en seis, de manera que las cuatro categorías suman 200.



C obtiene 176 aciertos y G obtiene 188, equivalentes a 88 % y 94 %; la diferencia neta es de 12 expedientes, o seis puntos porcentuales, pero la lectura pareada añade algo importante: G corrige 18 fallos de C y a la vez introduce seis fallos en casos que C resolvía.

Si los seis fallos exclusivos de G incluyen un tipo de excepción muy costosa, la diferencia agregada puede ocultar una regresión relevante; por eso proponemos conservar el expediente, la clase de error y el motivo de decisión, en vez de guardar solamente el porcentaje final.

Los números de esta demostración no acreditan significación estadística ni generalización a la carga futura; una evaluación real tendría que definir cómo se seleccionaron los casos, cuánto varían las respuestas, qué incertidumbre acompaña las tasas y qué volumen de evidencia permite tomar la decisión con el riesgo tolerado.

Qué registrar en la prueba

Antes de ejecutar, conviene escribir qué debe contener una respuesta aceptable, qué errores la invalidan y cuándo corresponde abstenerse; después puede registrarse cada

criterio por separado para distinguir una mejora sustantiva de una preferencia estilística o de una respuesta más persuasiva.

La comparación de revisión humana necesita incluir a quienes harán el trabajo y evitar que conocer la identidad del modelo determine de antemano la valoración; cuando resulte viable, ocultar esa identidad y alternar el orden de las respuestas ayuda a reducir una fuente de preferencia, sin convertir el procedimiento en garantía de imparcialidad.

Un conjunto de casos puede destinarse a ajustar instrucciones y otro mantenerse para comprobar la selección final; utilizar reiteradamente los mismos expedientes para diseñar la solución y anunciar su rendimiento favorece una conclusión demasiado dependiente de esos ejemplos, aunque no se hayan modificado los pesos del modelo.

También deben registrarse la versión del servicio y sus parámetros accesibles, el presupuesto de inferencia, las herramientas habilitadas y las fechas; si un proveedor modifica el sistema sin permitir fijar una versión, esa



limitación debe acompañar la comparación y motivar nuevas verificaciones cuando cambien los resultados.

Fallos que el tamaño no corrige por sí solo

Si el documento no llegó al sistema, una mayor capacidad de generación no crea una base documental legítima; el modelo podría completar huecos con información plausible, pero la intervención adecuada consiste en recuperar el material faltante y comprobar que la aplicación lo presenta correctamente.

Si una tabla perdió columnas durante la extracción, la comparación debe comenzar por revisar esa transformación; un modelo mayor quizá tolere algunos defectos de entrada, pero no conviene contratar capacidad adicional sin saber si una corrección determinista del extractor resolvería el problema con mayor previsibilidad.

Si el resultado requiere una suma exacta de importes disponibles, una calculadora o un programa verificable puede ser el componente apropiado; el modelo puede ayudar a identificar datos y explicar el resultado, aunque la

selección de tamaño no debería sustituir el diseño de una operación comprobable.

Cuando dos instrucciones del proceso se contradicen, tampoco basta con aumentar parámetros: alguien debe resolver qué criterio gobierna y qué excepciones están autorizadas; elegir una respuesta aparentemente coherente no demuestra que la aplicación haya resuelto correctamente una decisión organizativa que nunca se definió.

Estas situaciones no prueban que la capacidad del modelo sea irrelevante, sino que la mejora depende de identificar el origen del fallo; un diagnóstico que distingue extracción, recuperación, generación y validación permite invertir en el componente que realmente condiciona el resultado.

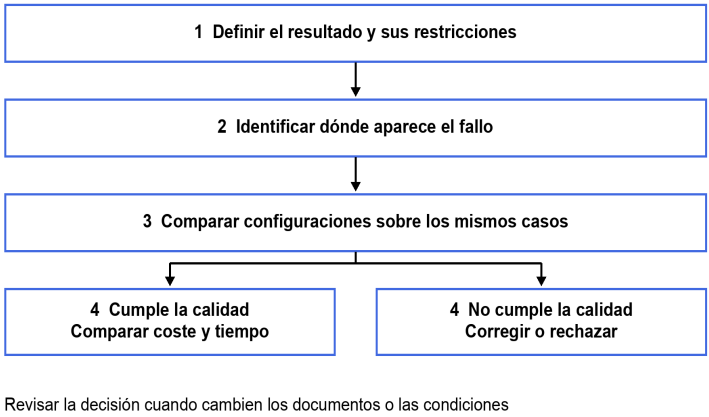
Un procedimiento de selección revisable

Proponemos comenzar por la unidad de trabajo y sus restricciones: qué entra, qué debe salir, qué calidad es indispensable, cuánto tiempo puede tardar y qué decisiones permanecen bajo responsabilidad humana; después corresponde seleccionar unas pocas

configuraciones plausibles y compararlas sobre casos pertinentes.

La Figura 1 resume la lógica de decisión; se trata de un diagrama conceptual propio, cuyas relaciones indican pasos de comprobación y no probabilidades ni resultados experimentales.

Figura 1. *Del problema observado a una elección comprobable*



Nota. Elaboración propia; el rechazo de una alternativa puede conducir a corregir el flujo o a solicitar mejores datos antes de repetir la evaluación.



El primer filtro debe ser el cumplimiento de restricciones que no pueden compensarse con una respuesta mejor escrita; si una opción no permite el tratamiento autorizado de los documentos o no puede operar en el entorno requerido, su desempeño en otras tareas no resuelve esa incompatibilidad.

Entre las alternativas viables, conviene comparar calidad inicial, calidad después de revisión, tiempo de entrega y recursos consumidos; una opción puede justificar un coste de ejecución mayor si reduce correcciones importantes, mientras otra puede resultar suficiente para tareas repetitivas que no aprovechan la capacidad adicional.

La decisión debe describir su alcance con precisión: «esta configuración cumple estos criterios sobre estos tipos de expediente» es una afirmación más útil que «este es el mejor modelo»; la primera permite identificar cuándo cambian las condiciones y la segunda tiende a ocultarlas.

También debe quedar definido qué observación obligaría a reconsiderar la elección, como una nueva clase de documentos, un incremento material de fallos, cambios de versión o una variación importante del volumen; revisar

con motivo no significa reemplazar continuamente una solución que funciona ni conservarla por inercia cuando sus supuestos dejan de cumplirse.

Objeciones que merece la pena conservar

Puede objetarse que evaluar consume tiempo y que elegir el modelo mayor resulta más sencillo; esa simplificación puede ser razonable para una exploración breve y de pocas consecuencias, pero resulta menos defendible cuando el sistema procesará un volumen elevado o cuando un error ocasionaría trabajo adicional significativo.

También puede objetarse que un modelo amplio ofrece capacidades futuras que una prueba estrecha no captura; esa flexibilidad tiene valor potencial, aunque debe distinguirse de beneficios ya comprobados y describirse mediante escenarios de uso plausibles, sin atribuir a necesidades futuras indeterminadas una ventaja económica segura.

Una tercera objeción sostiene que la alternativa pequeña siempre protege mejor el control operativo; el despliegue local puede aportar ciertas posibilidades, pero tamaño,



ubicación y condiciones de tratamiento son características distintas, por lo que la evaluación necesita verificar la configuración real y no deducir privacidad o seguridad de un conteo de parámetros.

Por último, es posible que ninguna alternativa domine: una puede ser mejor para documentos difíciles y otra para respuestas rápidas, con costes que cambian según el volumen; reconocer ese intercambio permite asignar funciones de forma explícita, siempre que la distribución se evalúe y no se suponga correcta por el simple hecho de combinar modelos.

Conclusión

Un modelo más grande mejora nuestro trabajo cuando su capacidad adicional resuelve dificultades relevantes y compensa los recursos necesarios para obtener un resultado aceptable; las investigaciones examinadas muestran que esa relación depende también del entrenamiento, los datos, la arquitectura y el presupuesto de inferencia.

Los casos hipotéticos explican por qué el precio de ejecución puede resultar engañoso y por qué distribuir solicitudes no garantiza ahorros compatibles con la calidad requerida; la selección debe conservar una relación comprobable entre el fallo observado, la intervención elegida y la mejora obtenida, incluso cuando eso conduzca a preferir un modelo menor o a corregir otro componente del proceso.

Referencias

Frantar, E., Ashkboos, S., Hoefler, T., y Alistarh, D. (2023). *GPTQ: Accurate post-training quantization for generative pre-trained transformers* [GPTQ: Cuantización precisa posterior al entrenamiento para transformers generativos preentrenados] (Versión 2)

[Prepublicación]. arXiv. https://doi.org/10.48550/arXiv.2210.17323

Hinton, G., Vinyals, O., y Dean, J. (2015). *Distilling the knowledge in a neural network* [Destilación del conocimiento en una red neuronal] (Versión 1) [Prepublicación]. arXiv. https://doi.org/10.48550/arXiv.1503.02531

Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., de Las Casas, D., Hendricks, L. A., Welbl, J., Clark, A., Hennigan, T., Noland, E., Millican, K., van den Driessche, G., Damoc, B., Guy, A., Osindero, S., Simonyan, K., Elsen, E., … Sifre, L. (2022). *Training compute-optimal large language models* [Entrenamiento de grandes modelos de lenguaje con cómputo óptimo] (Versión 1) [Prepublicación]. arXiv. https://doi.org/10.48550/arXiv.2203.15556

Jiang, A. Q., Sablayrolles, A., Roux, A., Mensch, A., Savary, B., Bamford, C., Chaplot, D. S., de las Casas, D., Bou Hanna, E., Bressand, F., Lengyel, G., Bour, G., Lample, G., Renard Lavaud, L., Saulnier, L., Lachaux, M.-A., Stock, P., Subramanian, S., Yang, S., … El Sayed, W. (2024). *Mixtral of experts* [Mixtral de expertos] (Versión 1) [Prepublicación]. arXiv. https://doi.org/10.48550/arXiv.2401.04088

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., y Amodei, D. (2020). *Scaling laws for neural language models* [Leyes de escalamiento para modelos neuronales de lenguaje] (Versión 1) [Prepublicación]. arXiv. https://doi.org/10.48550/arXiv.2001.08361

Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., Zhang, Y., Narayanan, D., Wu, Y., Kumar, A., Newman, B., Yuan, B., Yan, B., Zhang, C., Cosgrove, C., Manning, C. D., Ré, C., Acosta-Navas, D., Hudson, D. A., … Koreeda, Y. (2022). *Holistic evaluation of language models* [Evaluación integral de modelos de lenguaje] (Versión 1) [Prepublicación]. arXiv. https://doi.org/10.48550/arXiv.2211.09110

Sardana, N., Portes, J., Doubov, S., y Frankle, J. (2024). Beyond Chinchilla-optimal: Accounting for inference in language

model scaling laws [Más allá del óptimo de Chinchilla: Incorporación de la inferencia en las leyes de escalamiento de modelos de lenguaje]. En R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, y F. Berkenkamp (Eds.), *Proceedings of the 41st International Conference on Machine Learning* (Vol. 235, pp. 43445–43460). PMLR. https://proceedings.mlr.press/v235/sardana24a.html

Snell, C., Lee, J., Xu, K., y Kumar, A. (2024). *Scaling LLM test-time compute optimally can be more effective than scaling model parameters* [Escalar óptimamente el cómputo de un modelo de lenguaje al responder puede ser más eficaz que aumentar sus parámetros] (Versión 1) [Prepublicación]. arXiv. https://doi.org/10.48550/arXiv.2408.03314

