INGENIERÍA DE

ÍNDICE

OBJETIVOS

Esta guía tiene como objetivo principal capacitar a los alumnos de posgrado para dominar la ingeniería de prompts, desde sus fundamentos hasta técnicas avanzadas como el uso de modelos razonadores. Se busca que aprendan a optimizar la interacción con los LLM para generar respuestas de alta calidad en contextos académicos, gestionar sus limitaciones y aplicar estas herramientas de forma efectiva y crítica en el ámbito educativo.

I. DEFINICIÓN Y RELEVANCIA DE LA INGENIERÍA DE PROMPTS

I. DEFINICIÓN Y RELEVANCIA DE LA INGENIERÍA DE PROMPTS

La ingeniería de prompts se define como una nueva disciplina centrada en la optimización de las instrucciones (prompts) que se proporcionan a una inteligencia artificial para lograr los resultados deseados. Constituye la interfaz a través de la cual los usuarios humanos interactúan con los grandes modelos de lenguaje (LLM) [1]. La eficacia de los LLM en aplicaciones del mundo real depende intrínsecamente del diseño de prompts efectivos [1]. Se ha demostrado que los prompts diseñados por expertos pueden mejorar significativamente el comportamiento de los LLM [1].

audio_001

I. DEFINICIÓN Y RELEVANCIA DE LA INGENIERÍA DE PROMPTS

La consolidación de esta técnica como disciplina formal indica que la interacción efectiva con la IA no es trivial, sino que exige un conocimiento especializado. El hecho de que los prompts elaborados por expertos mejoren sustancialmente el rendimiento de los LLM implica una creciente demanda de profesionales que dominen esta área. En el ámbito educativo, esto significa que la mera utilización de los LLM es insuficiente; tanto educadores como estudiantes deben convertirse en usuarios competentes en sus interacciones con la IA, aprovechando así todo su potencial.

II. FUNDAMENTOS DE LA CONSTRUCCIÓN DE PROMPTS

LA CONSTRUCCIÓN DE PROMPTS EFECTIVOS ES LA BASE DE UNA INTERACCIÓN FRUCTÍFERA CON LOS LLM, POR LO QUE ES ESENCIAL COMPRENDER SUS ELEMENTOS CONSTITUTIVOS Y LOS PRINCIPIOS QUE RIGEN SU OPTIMIZACIÓN.

ELEMENTOS ESENCIALES DE UN PROMPT

Un prompt es, en esencia, un conjunto de instrucciones que se le dan a una inteligencia artificial para indicarle la tarea que se desea que realice. Estos prompts pueden ser desde preguntas breves y simples hasta instrucciones complejas que ocupan varios párrafos. Para una utilización eficiente de los LLM, es necesario conocer los elementos que pueden conformar un prompt, aunque no todos deben estar siempre presentes.

II. FUNDAMENTOS DE LA CONSTRUCCIÓN DE PROMPTS

INSTRUCCIÓN

Es la parte específica del prompt que establece la consulta o petición principal dirigida al modelo. Representa el núcleo semántico del prompt y puede presentarse como una pregunta o una afirmación.
Por ejemplo, "¿En qué consiste el proceso de fotosíntesis?" o "Describe el proceso de la fotosíntesis".

II. FUNDAMENTOS DE LA CONSTRUCCIÓN DE PROMPTS

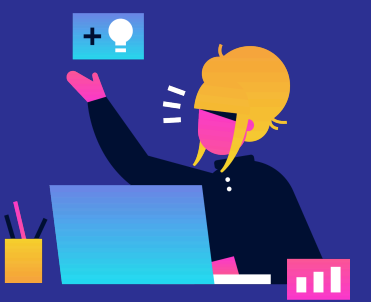
Escribe un tutorial	Explica detalladamente	Realiza un resumen	Compara
Identifica los errores en	Proporciona ejemplos	Analiza ventajas y desventajas	Crea un plan de estudio
Genera ideas para	Traduce el siguiente texto al español	Corrige la gramática y ortografía	Responde la siguiente pregunta
Completa la oración	Escribe un ensayo sobre	Desarrolla un guion para	Modifica el siguiente código para
Simplifica este concepto	Crea una lista de	Redacta un correo electrónico	Genera un informe sobre
Diseña una estrategia para	Inventa una historia sobre	Describe la imagen	Categoriza los siguientes elementos

audio_003

II. FUNDAMENTOS DE LA CONSTRUCCIÓN DE PROMPTS

Predice el resultado de	Argumenta a favor de	Argumenta en contra de	Sintetiza la información clave de
Expande este punto	Haz una lluvia de ideas sobre	Explora las implicaciones de	Formula una hipótesis
Evalúa la viabilidad de	Propón una solución para	Genera un eslogan para	Escribe una reseña de
Simula una conversación	Realiza un análisis FODA de	Prepara un discurso sobre	Concluye este texto
Recomienda recursos para	Explora diferentes perspectivas sobre	Crea un caso de estudio sobre	Resume los puntos principales de

CONTEXTO



Involucra información adicional relevante que se proporciona al modelo. Esta información previa, generalmente incorporada en los prompts iniciales de una cadena de diálogo, es fundamental para lograr una mayor especificidad en la respuesta.
Un ejemplo sería: "Estoy preparando una clase para alumnos universitarios de la carrera de Psicología, sobre la teoría de Freud respecto de la interpretación de los sueños, proporciona tres ejemplos sobre…".
Proporcionar contexto ayuda al LLM a construir un marco de referencia a partir de la información suministrada, lo que conduce a respuestas más pertinentes [2]. Además, especificar el tono deseado en el prompt puede mejorar el contexto, influyendo en cómo se presenta la información para satisfacer las necesidades de la audiencia [2].

INDICADOR DE SALIDA

Permite especificar el tipo o formato que se espera en la respuesta del modelo. Esto puede ser una lista de elementos, una solicitud de resumen, una tabla o cualquier otro formato específico.
Por ejemplo, "Genera una lista de eventos históricos en orden cronológico que expliquen…" o "Proporciona una explicación detallada en formato de ensayo". El indicador de salida puede colocarse al final del prompt. Especificar la forma de la salida asegura que la respuesta cumpla con necesidades y expectativas específicas, contribuyendo a la concisión del resultado [2].

II. FUNDAMENTOS DE LA CONSTRUCCIÓN DE PROMPTS

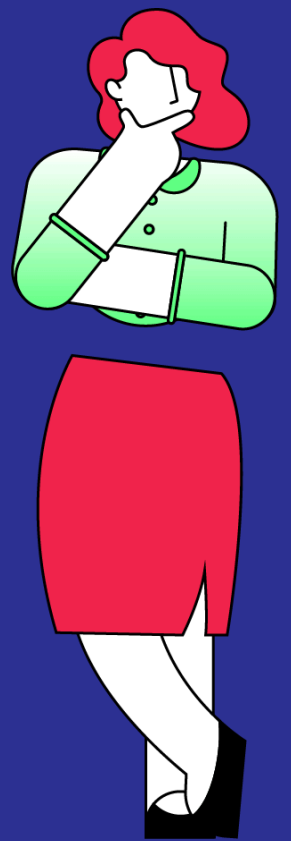
Cada componente cumple una función comunicativa específica. El énfasis en la provisión de contexto resalta que los LLM, a pesar de su vasto conocimiento, no son omniscientes y requieren una delimitación explícita para alinear sus respuestas con la intención del usuario. El indicador de salida, por su parte, demuestra que los usuarios no solo deben transmitir qué desean, sino también cómo debe presentarse, lo que sugiere un diálogo estructurado en lugar de uno de forma libre. Esto implica que dominar la ingeniería de prompts requiere comprender el "lenguaje" del LLM y sus mecanismos de procesamiento internos.

Entre los componentes también se puede considerar:

Inclusión de Ejemplos
En situaciones donde se busca un resultado altamente específico, la provisión de pares de ejemplos de entrada y salida puede orientar eficazmente al modelo. Este método se conoce como "few-shot learning" o aprendizaje con pocas muestras.

Restricciones o Limitaciones
La enunciación de parámetros restrictivos o exclusiones en la respuesta es tan relevante como la especificación de los elementos deseados. Por ejemplo: "No incluya terminología técnica especializada.", "Evite el empleo de superlativos.", "La lista no deberá exceder de cinco ítems."

II. FUNDAMENTOS DE LA CONSTRUCCIÓN DE PROMPTS



PRINCIPIOS BÁSICOS DE OPTIMIZACIÓN DE PROMPTS

audio_004

La optimización de prompts es una disciplina emergente que busca maximizar la eficiencia en el uso de los LLM. En la actualidad, estas técnicas incluyen métodos automatizados que, a menudo, aprovechan la capacidad de los propios LLM para refinar y adaptar los prompts [1]. Esto indica que los LLM también pueden actuar como agentes en su propia optimización, creando un ciclo de retroalimentación.

El Proceso Iterativo de Diseño de Prompts

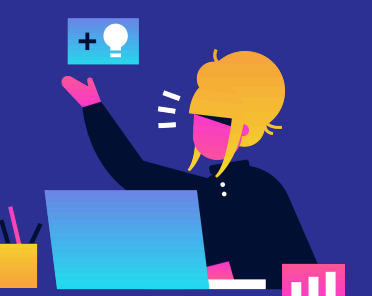
El diseño de prompts es sustancialmente un proceso iterativo. Se recomienda diseñar un prompt inicial y, de manera progresiva, añadir más elementos y contexto hasta llegar a la versión definitiva que produzca los mejores resultados. Es beneficioso probar diferentes estructuras con distintos contextos y datos, observando los resultados de cada iteración.

La ingeniería de prompts exitosa requiere pensamiento crítico, adaptabilidad y una disposición a aprender de cada interacción. Es una habilidad que mejora con la práctica y la reflexión, en lugar de un conjunto fijo de reglas.

ESPECIFICIDAD Y PRECISIÓN EN LA REDACCIÓN DE PROMPTS

audio_005

Es de suma importancia ser muy específico y preciso en la descripción de la tarea que se desea que el modelo realice. Cuanto más descriptivo y detallado sea el prompt, mejores serán los resultados. Un prompt claro y directo guía al modelo en la dirección correcta, lo que le permite generar respuestas más enfocadas y coherentes, y reduce la probabilidad de obtener resultados incorrectos o malinterpretados. Por el contrario, un enunciado demasiado amplio o vago generalmente conduce a respuestas superficiales y generalizadas.



II. FUNDAMENTOS DE LA CONSTRUCCIÓN DE PROMPTS

Un ejemplo ilustrativo de un prompt poco específico sería: "Explora el impacto de la tecnología en la sociedad." Este prompt carece de delimitación clara sobre los aspectos específicos de la tecnología y la sociedad que se esperan abordar, lo que lleva a respuestas generales. Una versión mejorada y más específica sería: "Analiza el impacto de las redes sociales en la interacción social y la formación de identidad en la era digital, considerando tanto los aspectos positivos como los desafíos éticos y de privacidad asociados". Para tareas extensas que involucran múltiples sub-tareas, es preferible dividirlas en partes más simples e incorporarlas en el diálogo en varios pasos, evitando así una complejidad excesiva en el diseño inicial del prompt.

La reiterada insistencia en la especificidad y precisión, a pesar de que los LLM son modelos extensos entrenados con vastas cantidades de datos, revela una paradoja. Aunque poseen un conocimiento muy amplio, los LLM requieren una entrada humana altamente restringida para funcionar de manera óptima. Un prompt vago a menudo conduce a resultados superficiales o engañosos, lo que sugiere que el comportamiento predeterminado del modelo es, con frecuencia, demasiado generalizado.

II. FUNDAMENTOS DE LA CONSTRUCCIÓN DE PROMPTS

ASIGNACIÓN DE ROLES A LA IA

La técnica de asignar un rol a la IA al redactar un prompt consiste en contextualizar la interacción al establecer una identidad específica para el modelo, generalmente la de un experto idóneo en el tema de la pregunta. Al indicarle al sistema cómo comportarse, cuál es su intención y su identidad, se le proporciona un marco de referencia que lo guía para generar respuestas relevantes desde una perspectiva más especializada. Esta técnica permite obtener respuestas más precisas, contextualmente apropiadas y coherentes.

Ejemplos comunes de asignación de roles incluyen: "Imagina que eres un chef de renombre. Proporciona una receta…", "Actúa como si fueras un profesor de historia. Explica detalladamente el contexto histórico…", o "Eres un entrenador personal. Diseña un programa de ejercicios…".

audio_006

ASIGNACIÓN DE ROLES A LA IA

audio_007

Asignar un rol a la IA es más que simplemente establecer un contexto; es un mecanismo potente para alinear su comportamiento e inyectar una persona específica. Al instruir al LLM sobre cómo debe comportarse, cuál es su intención y su identidad, el prompt programa eficazmente una persona temporal y especializada.

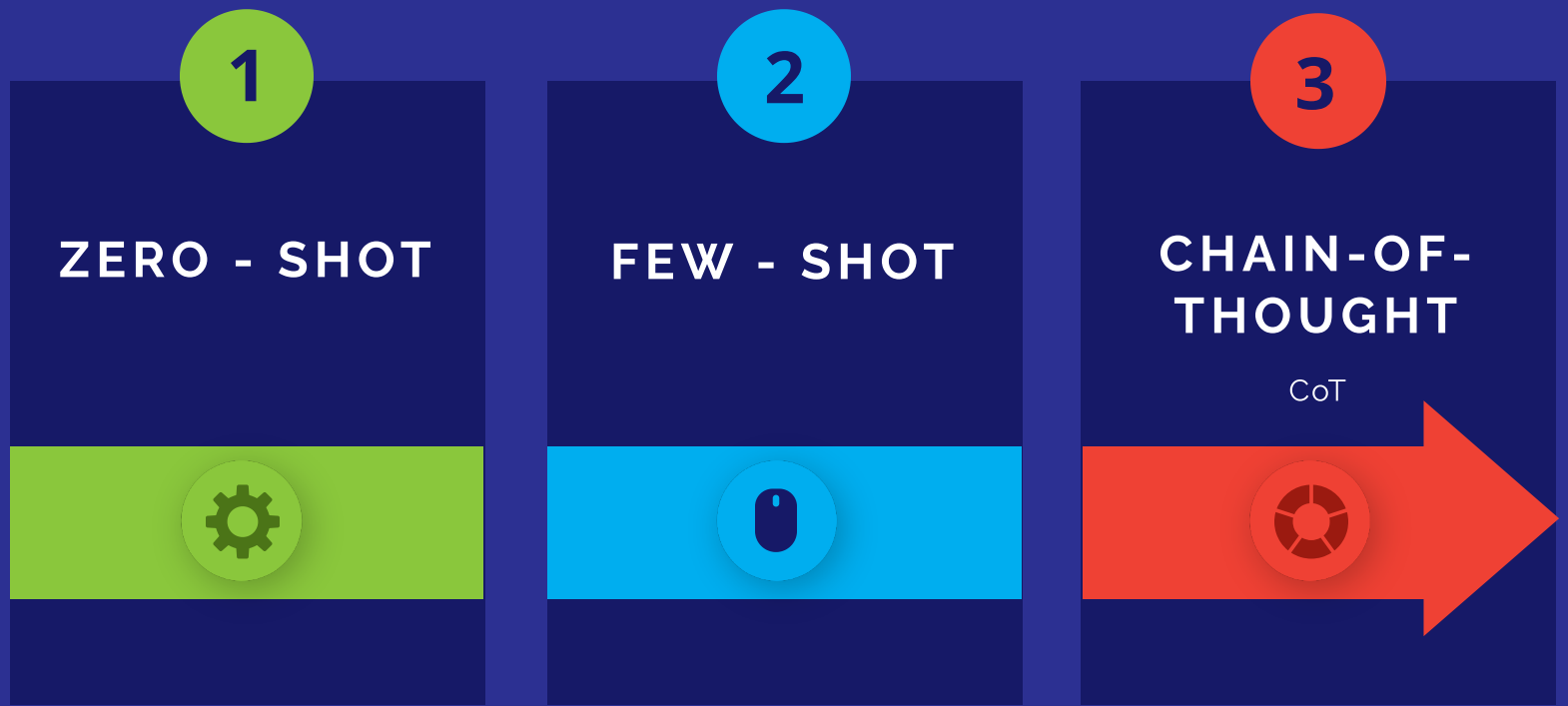
Identificación de la audiencia y adaptación del lenguaje:

Es importante considerar la audiencia a la que va dirigido el prompt. ¿Son estudiantes universitarios, expertos en un determinado campo o personas con conocimientos generales? Adaptar el lenguaje y el nivel de complejidad del prompt permitirá una comunicación más efectiva y comprensible para la audiencia específica.

III. TÉCNICAS DE PROMPTING

III. TÉCNICAS DE PROMPTING

Más allá de los fundamentos, existen técnicas de prompting avanzadas que permiten a los LLM abordar tareas de mayor complejidad. Estas estrategias, como el prompting zero-shot, few-shot y Chain-of-Thought (CoT), sirven para mejorar la capacidad de los modelos en la realización de tareas específicas, especialmente aquellas que requieren razonamiento [3].



III. TÉCNICAS DE PROMPTING

1

ZERO - SHOT

La estrategia Zero-shot (cero disparos) se refiere a la capacidad de un modelo para realizar tareas sin ningún ejemplo de entrenamiento específico para esa tarea en particular.
Se implementa proporcionando una descripción general o un conjunto de instrucciones en el prompt inicial, indicando al modelo qué se espera que haga. Esta técnica es ideal cuando la tarea es relativamente sencilla y el conocimiento general del LLM es suficiente para generar resultados precisos.5Las mejores prácticas para el prompting zero-shot incluyen proporcionar instrucciones claras y concisas, y evitar tareas ambiguas o complejas donde el modelo podría necesitar más orientación [4].

La capacidad de un modelo para realizar una tarea sin ejemplos de entrenamiento específicos para esa tarea demuestra el notable poder de generalización implícita que reside en el pre-entrenamiento de los LLM. Pueden asistir de inmediato en una amplia gama de tareas sin necesidad de una afinación previa extensa, lo que resalta la importancia de dar instrucciones claras para guiar eficazmente esta amplia capacidad de generalización.

III. TÉCNICAS DE PROMPTING

2

FEW - SHOT

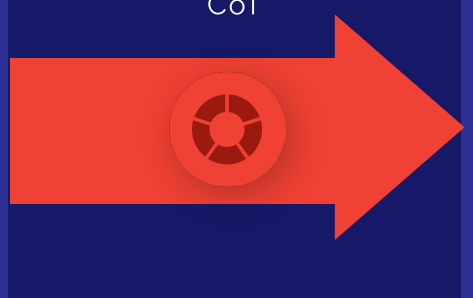
La estrategia de few-shot (pocos disparos) es similar a zero-shot, pero se diferencia en que se proporciona al modelo un conjunto reducido de ejemplos de entrenamiento dentro del propio prompt para la tarea específica. De esta manera, el modelo realiza tareas con una cantidad limitada de ejemplos, lo que resulta particularmente útil para escenarios donde no se dispone de grandes cantidades de datos etiquetados [5].
Cuando el prompt zero-shot no conduce a una respuesta adecuada, se recomienda proporcionar demostraciones o ejemplos en la instrucción, lo que orienta al modelo hacia una mejor respuesta. Las mejores prácticas para el prompting few-shot incluyen proporcionar ejemplos claros y representativos, y mantener la coherencia en el formato [4].
Esto es especialmente relevante para tareas que requieren resultados específicos, donde los ejemplos explícitos son más efectivos que las instrucciones abstractas.

III. TÉCNICAS DE PROMPTING

3

CHAIN-OF-THOUGHT

CoT

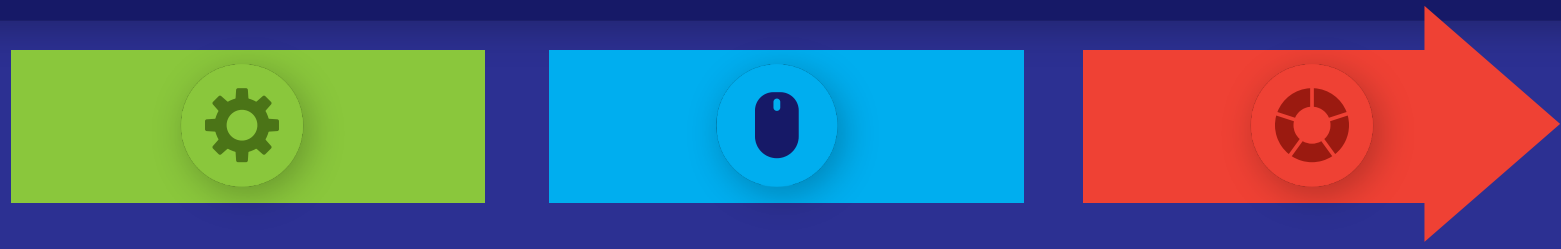


La estrategia Chain-of-Thought (CoT) o Cadena de Pensamientos es una técnica avanzada que mejora las capacidades de razonamiento de los LLM al desglosar tareas complejas en sub-pasos más simples. Implica utilizar una secuencia de tareas progresivamente más difíciles o desafiantes, donde el modelo comienza con tareas más simples y avanza hacia otras más complejas, construyendo conocimiento y habilidades en el proceso. Al instruir a los LLM para que resuelvan problemas paso a paso, se hace explícito y verificable el proceso de pensamiento del modelo [6]. Esto es importante porque los LLM no especializados en razonamiento aún enfrentan dificultades para realizar tareas de razonamiento avanzadas, que implican comprender y procesar información de manera lógica, realizar inferencias y aplicar reglas y principios para llegar a conclusiones válidas.

Para los modelos comunes, se puede lograr un resultado aceptable simplemente indicando al modelo que "Piense paso a paso". Esta capacidad se considera una habilidad emergente, es decir, una capacidad que aparece a medida que el tamaño o la complejidad del modelo aumenta [6].

Al obligar al modelo a descomponer problemas en pasos de razonamiento intermedios, haciendo explícito y verificable el proceso de pensamiento, CoT permite a los usuarios observar y, si es necesario, depurar la lógica interna del LLM.

audio_008

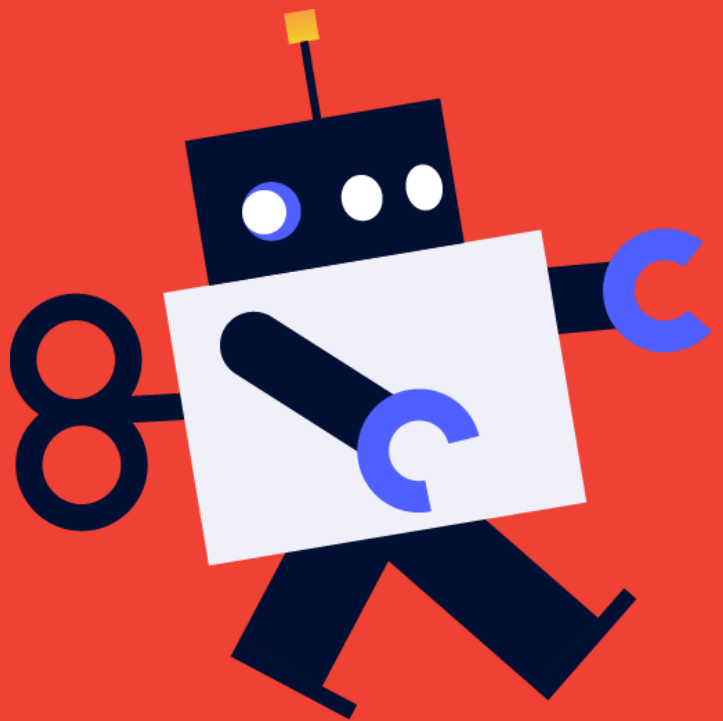


MANEJO DE ERRORES Y RETROALIMENTACIÓN AL MODELO

Cuando un modelo de lenguaje responde de manera incoherente o equivocada, se pueden adoptar dos enfoques principales para obtener mejores resultados: indicar el error o volver a redactar el prompt.

Indicar el Error: Si se detecta que la respuesta es incorrecta o incoherente, se puede proporcionar retroalimentación al modelo indicando explícitamente el error y ofreciendo una explicación adicional que aclare o corrija la información incorrecta. Por ejemplo: "No es correcto. La respuesta que proporcionaste es falsa. Te voy a dar más contexto para que puedas generar una respuesta correcta. [...]".

Volver a Redactar el Prompt: Si se considera que el prompt original pudo haber sido ambiguo, poco claro o insuficiente para orientar adecuadamente al modelo, se puede optar por reformular e introducir nuevamente el prompt, corrigiendo sus deficiencias, sin señalar al modelo que ha cometido un error. Un ejemplo sería: "Voy a reformular la pregunta para brindar más detalles. Responde considerando los siguientes aspectos: [...]".



IV. MODELOS RAZONADORES

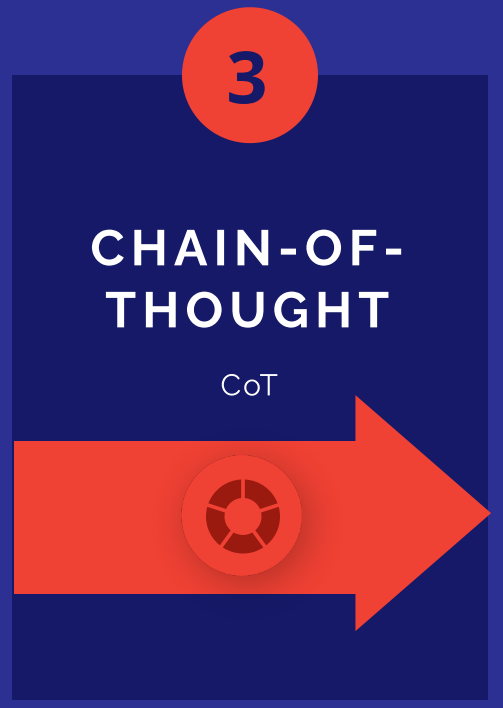
IV. MODELOS RAZONADORES

El razonamiento es una capacidad cognitiva fundamental que permite la inferencia lógica, la resolución de problemas y la toma de decisiones. Con el rápido avance de los LLM, el razonamiento ha surgido como una capacidad clave que distingue a los sistemas de IA avanzados de los modelos convencionales [7].

IV. MODELOS RAZONADORES

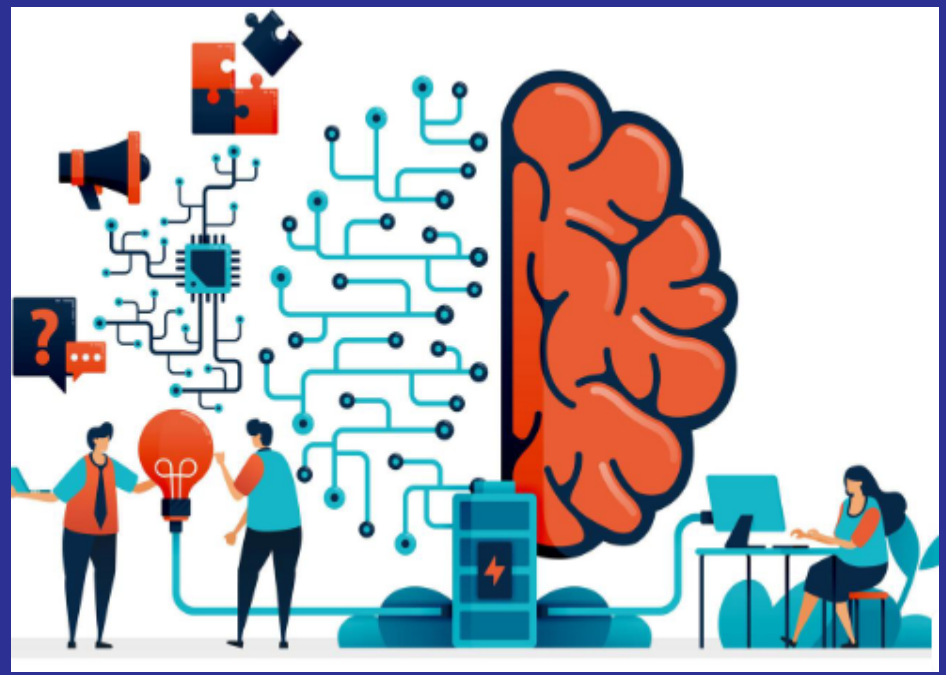
La metodología **Chain-of-Thought (CoT)** ha revolucionado la forma en que los Grandes Modelos de Lenguaje (LLM) abordan el razonamiento complejo, permitiendo que descompongan los problemas en pasos intermedios y hagan explícito su proceso de pensamiento [8]. El éxito de CoT se ve afectado por factores como la probabilidad y la memorización, lo que sugiere que el "razonamiento" de los LLM está más ligado a patrones estadísticos que a la deducción lógica, y que la estructura y el formato de los ejemplos de demostración son más cruciales que su validez [8].

A pesar de los avances, los LLM aún enfrentan desafíos para ir más allá del emparejamiento de patrones y lograr un razonamiento más robusto y generalizable, similar al humano [8]. La divergencia entre el razonamiento humano y el de los LLM se manifiesta en la dificultad de estos últimos para generalizar más allá de los datos de entrenamiento, aplicar principios lógicos de manera consistente, manejar escenarios novedosos y integrar múltiples pasos de razonamiento, mostrando una degradación del rendimiento ante variaciones mínimas en la presentación del problema [8]. Esto indica que la "comprensión" de los LLM no es tan robusta o transferible como el razonamiento humano.



IV. MODELOS RAZONADORES

Los Modelos de Razonamiento a Gran Escala (LRM), como o3 de OpenAI, QwenQwQ y DeepSeek-R1, representan un avance significativo en el razonamiento generativo [9]. Estos modelos, que utilizan aprendizaje por refuerzo a gran escala, pueden recuperar y refinar información dinámicamente, generando procesos de pensamiento coherentes y de múltiples pasos [9]. A diferencia de los LLM que solo dan una respuesta final, los LRM resuelven problemas complejos (como acertijos, desafíos de codificación y problemas matemáticos) generando o gestionando internamente pasos intermedios de "pensamiento", lo que mejora la coherencia lógica y la interpretabilidad [10, 9].



V. OPTIMIZACIÓN Y FORMATO AVANZADO DE PROMPTS

V. OPTIMIZACIÓN Y FORMATO AVANZADO

PAUTAS PARA LA ESTRUCTURA DE LA SALIDA Y LA INDICACIÓN DE ESTILO

Mientras que la mayoría de la investigación inicial en ingeniería de prompts se centró en el contenido, estudios recientes han puesto de manifiesto la importancia crítica del formato del prompt. Este apartado explora cómo la estructura y presentación de un prompt pueden impactar significativamente el rendimiento del LLM y presenta enfoques avanzados para su optimización.

Además de optimizar el prompt de entrada, es fundamental guiar al LLM en la generación de la salida deseada, tanto en su estructura como en su estilo.

V. OPTIMIZACIÓN Y FORMATO AVANZADO

PAUTAS DE ESTRUCTURA

Para evitar respuestas excesivamente breves, se puede indicar al modelo el tipo de estructura que se espera. Un prompt bien diseñado debe proporcionar instrucciones específicas sobre el formato y el contenido. Por ejemplo, utilizar el término 'ensayo' en el prompt, como en "Escribe un ensayo altamente detallado con secciones de introducción, cuerpo y conclusión que responda a lo siguiente: ¿Cuáles son los problemas…", hace más probable que el LLM genere respuestas coherentes y estructuradas, ya que comprende la definición de un ensayo?

V. OPTIMIZACIÓN Y FORMATO AVANZADO

INDICACIÓN DE ESTILO

audio_010

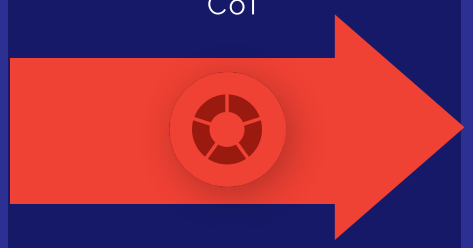
Es posible solicitar a la IA que adopte un estilo de lenguaje específico durante la conversación. Mediante esta solicitud, se pueden obtener respuestas que reflejen el tono, la terminología y las características propias de un estilo particular. Si no se especifica el estilo, la IA probablemente generará uno o dos párrafos cortos como respuesta. Se puede solicitar que el estilo sea concordante con algún rol previamente asignado al modelo, por ejemplo: “Escribe con el estilo y la terminología de un experto en química, doctorado y con más de 20 años de experiencia. Explica con ejemplos detallados el tema […]”. También es factible solicitar a la IA que adopte un estilo de escritura específico de un reconocido autor o diversos estilos como poético, irónico, creativo, informal, académico, o para ser comprendido por un niño de 5 años.

V. OPTIMIZACIÓN Y FORMATO AVANZADO

3

CHAIN-OF-THOUGHT

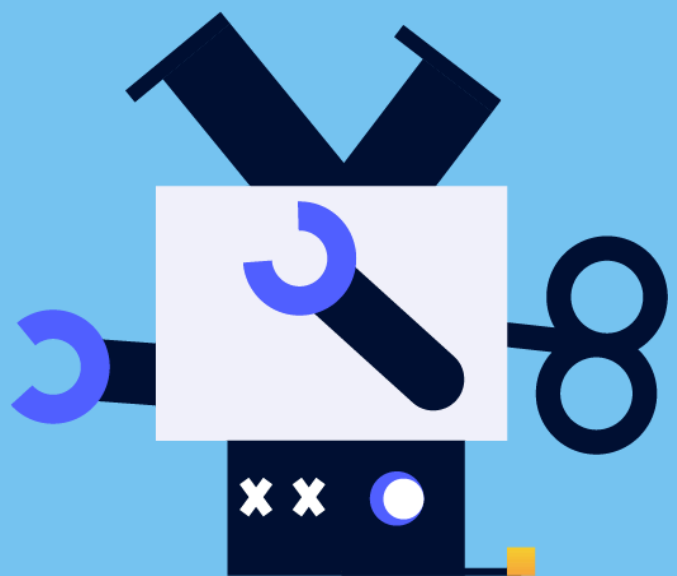
CoT



La estrategia Chain-of-Thought (CoT) o Cadena de Pensamientos es una técnica avanzada que mejora las capacidades de razonamiento de los LLM al desglosar tareas complejas en sub-pasos más simples. Implica utilizar una secuencia de tareas progresivamente más difíciles o desafiantes, donde el modelo comienza con tareas más simples y avanza hacia otras más complejas, construyendo conocimiento y habilidades en el proceso. Al instruir a los LLM para que resuelvan problemas paso a paso, se hace explícito y verificable el proceso de pensamiento del modelo [6]. Esto es importante porque los LLM no especializados en razonamiento aún enfrentan dificultades para realizar tareas de razonamiento avanzadas, que implican comprender y procesar información de manera lógica, realizar inferencias y aplicar reglas y principios para llegar a conclusiones válidas.

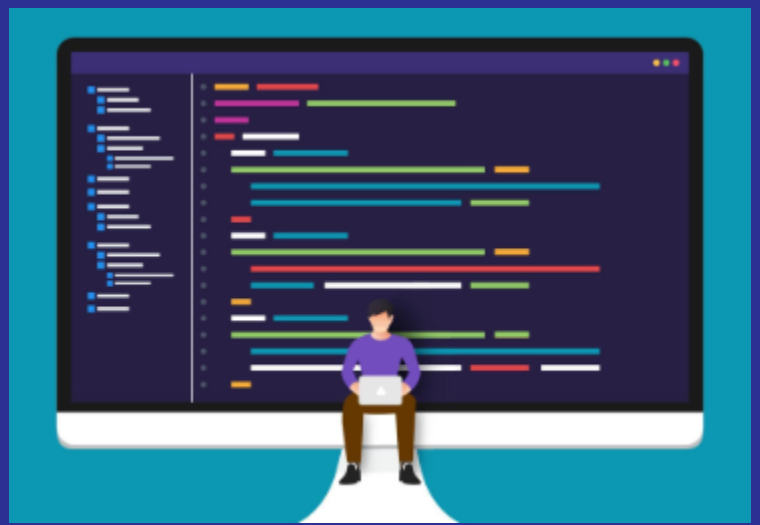
Para los modelos comunes, se puede lograr un resultado aceptable simplemente indicando al modelo que "Piense paso a paso". Esta capacidad se considera una habilidad emergente, es decir, una capacidad que aparece a medida que el tamaño o la complejidad del modelo aumenta [6].

Al obligar al modelo a descomponer problemas en pasos de razonamiento intermedios, haciendo explícito y verificable el proceso de pensamiento, CoT permite a los usuarios observar y, si es necesario, depurar la lógica interna del LLM.



VI. CONSIDERACIONES PRÁCTICAS Y LIMITACIONES DE LOS LLM

Generación de código



Los LLM son extremadamente efectivos en la generación de código, lo que resulta muy útil en el ámbito de la programación. Estos modelos están entrenados con grandes conjuntos de datos de código fuente en variados lenguajes, lo que les permite aprender patrones y estructuras comunes en la programación.
La capacidad de los LLM para generar código se extiende más allá de la simple replicación. Pueden asistir en la autocompletado de código, la corrección de errores, la refactorización, y la creación de fragmentos de código basados en descripciones en lenguaje natural. Esto optimiza el flujo de trabajo de los desarrolladores, reduce el tiempo de desarrollo y minimiza la aparición de errores.

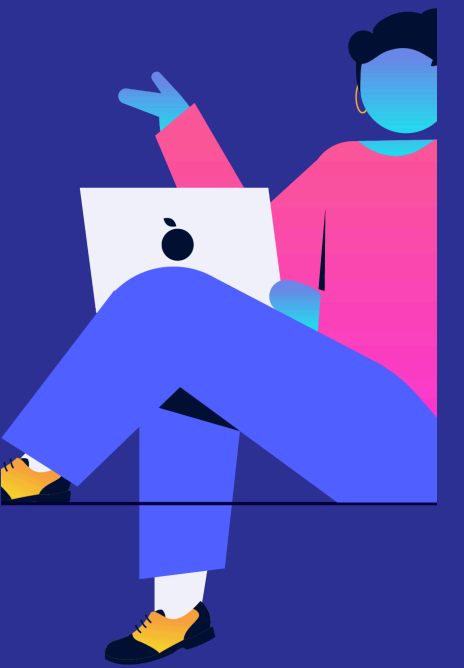
Solicitud de Feedback a la IA

La inteligencia artificial (IA) puede funcionar como una herramienta valiosa para la retroalimentación y mejora de textos. Al presentar un escrito a un modelo de IA, es posible obtener sugerencias para corregir errores gramaticales, inconsistencias, contradicciones, problemas de estilo o estructura, así como recomendaciones para la claridad del contenido y la precisión en la elección de palabras. Esta capacidad transforma al Modelo de Lenguaje Grande (LLM) de un simple generador de contenido a un socio metacognitivo.

La IA puede analizar el contenido creado por humanos y ofrecer comentarios sobre errores, inconsistencias o mejoras estilísticas. Esta función fomenta un entorno de aprendizaje colaborativo, brindando a los estudiantes retroalimentación inmediata y personalizada que puede potenciar sus habilidades de escritura y análisis crítico. De este modo, la IA pasa de producir contenido a criticarlo y mejorarlo, replicando los procesos de revisión por pares o la retroalimentación de un instructor.

VI. CONSIDERACIONES PRÁCTICAS Y LIMITACIONES

SOLICITUD DE FEEDBACK A LA IA



La Inteligencia Artificial se convierte en una herramienta invaluable para la retroalimentación y mejora de textos. Al analizar un escrito, la IA puede detectar y sugerir mejoras en errores gramaticales, inconsistencias, contradicciones, problemas de estilo o estructura, así como optimizar la claridad del contenido y la precisión léxica. Esta funcionalidad transforma a los LLM de simples generadores a socios metacognitivos.

Esta capacidad de la IA fomenta un entorno de aprendizaje colaborativo, ofreciendo a los usuarios —especialmente estudiantes— retroalimentación inmediata y personalizada sobre su trabajo. Esto no solo eleva la calidad de sus producciones, sino que también perfecciona sus habilidades de escritura y pensamiento crítico. En esencia, la dinámica de la IA pasa de solo crear contenido a criticarlo y mejorarlo, imitando procesos de revisión por pares o la orientación de un instructor.

VI. CONSIDERACIONES PRÁCTICAS Y LIMITACIONES

Puntos Débiles Comunes

A pesar de los avances significativos en los últimos años, los LLM presentan limitaciones estructurales y funcionales que deben ser comprendidas para un uso responsable y crítico, especialmente en contextos académicos, científicos y profesionales. A continuación, se exponen algunos de sus principales puntos débiles.

1. Información Desactualizada y Sincronización con Internet

Los LLM son entrenados a partir de grandes volúmenes de datos disponibles públicamente, con un corte temporal específico. Esto significa que el conocimiento que poseen está restringido a la información disponible hasta la fecha de su último entrenamiento. Aunque algunos modelos, como los más recientes desarrollados por OpenAI (por ejemplo, GPT-4 Turbo u o3), han sido entrenados hasta fechas recientes (2024 o 2025), la fecha de corte sigue siendo una limitación estructural, ya que el modelo no aprende continuamente por sí solo después de ser entrenado.

¿Tienen conexión a Internet en tiempo real?

De forma predeterminada, los LLM no están conectados permanentemente a Internet en tiempo real. Esto significa que no realizan búsquedas activas ni consultan información externa al generar cada respuesta. En lugar de ello, funcionan como un sistema cerrado basado en el conocimiento aprendido durante el entrenamiento.

Sin embargo, algunos modelos avanzados pueden acceder a herramientas auxiliares —como módulos de búsqueda web— cuando están integrados en plataformas que lo permiten. Por ejemplo, el sistema de ChatGPT puede usar una herramienta denominada "web", que realiza consultas en motores de búsqueda y permite al modelo recuperar información actualizada, leer enlaces y sintetizar contenidos recientes. Esta capacidad no forma parte del modelo central, sino que constituye una herramienta externa que puede ser activada cuando se requiere y bajo ciertas condiciones, como en el caso de tareas explícitas que demandan información actual o verificación de hechos.

¿Tienen conexión a Internet en tiempo real?

El modelo no "navega la web" en tiempo real de forma autónoma ni constante mientras genera cada respuesta, pero puede incorporar capacidades de consulta externa si está conectado a herramientas de búsqueda que funcionan como extensiones específicas.

¿Por qué esto es relevante?

Porque no todos los modelos de lenguaje actuales (ni siquiera todos los de OpenAI) tienen esta capacidad. Por ejemplo, un LLM descargado para uso local (como LLaMA en una PC) nunca tendrá acceso a Internet en tiempo real, a menos que se le integre explícitamente. En entornos académicos, clínicos o legales, es crítico saber si el modelo accede a fuentes actuales o no, porque de eso depende la validez de la información.

VI. CONSIDERACIONES PRÁCTICAS Y LIMITACIONES

2. CITAS BIBLIOGRÁFICAS INEXACTAS O FABRICADAS

Una de las limitaciones más notorias en el uso académico de los LLM es su incapacidad para citar fuentes con precisión, especialmente cuando no se integran con herramientas de búsqueda web o bases de datos científicas. Los modelos pueden generar citas que aparentan ser reales —con autores, títulos y fechas plausibles—, pero que en realidad no corresponden a documentos existentes, lo que se conoce como alucinación referencial. Esto se debe a que el modelo no recuerda textos exactos ni enlaces verificables, sino que genera texto conforme a patrones estadísticos aprendidos.
Para trabajos académicos, investigaciones científicas o contextos que requieran referencias rigurosas, se recomienda utilizar herramientas específicas como Semantic Scholar, Scite, ResearchRabbit o Consensus, o bien trabajar en conjunto con LLM que estén conectados a buscadores académicos o bases de datos verificadas.

VI. CONSIDERACIONES PRÁCTICAS Y LIMITACIONES

3. DIFICULTADES CON CÁLCULOS MATEMÁTICOS COMPLEJOS

Si bien los LLM han demostrado avances en la resolución de problemas matemáticos simples o simbólicos, sus limitaciones se hacen evidentes frente a cálculos complejos que requieren razonamiento formal, manipulación simbólica o múltiples pasos lógicos encadenados.
Esto ocurre porque los modelos generan texto a partir de predicciones lingüísticas, no a partir de ejecuciones computacionales reales. Por ello, pueden simular razonamientos, pero cometen errores en algebra avanzada, cálculo diferencial, teoría de probabilidades o estadística matemática.

VI. CONSIDERACIONES PRÁCTICAS Y LIMITACIONES

3. DIFICULTADES CON CÁLCULOS MATEMÁTICOS COMPLEJOS

Actualmente existen dos líneas de mejora:

• Modelos razonadores como DeepSeek R1 o OpenAI o3, que han sido afinados para mostrar habilidades de razonamiento lógico-matemático más robustas, mediante estrategias de descomposición de problemas, justificación paso a paso y entrenamiento especializado en tareas de razonamiento. Sin embargo, estos modelos no ejecutan cálculos reales: razonan lingüísticamente sobre procedimientos matemáticos, pero sin acceso a motores de cómputo formales.

• Modelos con ejecución externa, como GPT-4 Turbo con la herramienta "Python" o "Advanced Data Analysis", que integran entornos de ejecución reales (por ejemplo, intérpretes de Python). En estos casos, el modelo puede generar código, ejecutarlo en tiempo real y analizar sus propios resultados, lo que incrementa significativamente la precisión en tareas matemáticas o estadísticas complejas.
Para cálculos avanzados se recomienda utilizar LLM con herramientas de ejecución activa y verificar siempre los resultados con entornos especializados como Wolfram Alpha, Mathematica, MATLAB, etc.

4. Contexto Limitado y Pérdida de Coherencia

Los LLM procesan la información dentro de una ventana de contexto finita, que varía según el modelo y llega actualmente a 2 millones de tokens, en Gemini 2.5 Pro. Esto significa que no pueden “recordar” todo lo que se ha dicho indefinidamente. A medida que una conversación se extiende, los tokens más antiguos salen de la ventana de atención, y el modelo pierde acceso directo a esa información.

Gestión de Conversaciones Largas: La Ventana de Contexto y la Gestión de Memoria

Los LLM operan basándose en una ventana contextual limitada, lo que significa que solo pueden considerar una cantidad determinada de palabras o tokens anteriores en su contexto. A medida que se añade más texto en una conversación, los tokens más antiguos quedan fuera de esta ventana y, por lo tanto, tienen menos peso en la generación de respuestas. Esta limitación de memoria conduce a una disminución en la calidad de la conversación, ya que el modelo "olvida" información relevante o detalles específicos mencionados anteriormente, lo que afecta la coherencia de la interacción.

4. Contexto Limitado y Pérdida de Coherencia

Una estrategia efectiva para abordar este problema es solicitar periódicamente al modelo que genere un resumen o sumario de la conversación hasta ese momento. Al hacer esto, se fuerza al modelo a recordar toda la interacción y a restablecer la importancia de todas las etapas de la conversación, ayudando a mantener una visión general y a evitar la pérdida de detalles importantes a medida que la conversación avanza. Si bien la sumarización periódica es una solución manual, la aparición de agentes LLM representa un cambio fundamental para superar esta limitación. Estos sistemas agénticos están diseñados para retener y recuperar información relevante durante interacciones prolongadas.

Esto afecta especialmente en tareas que requieren mantener múltiples hilos argumentales, consistencia narrativa o seguimiento preciso de instrucciones en diálogos prolongados. Aunque existen mejoras como la memoria conversacional persistente (en algunos entornos), esta no es perfecta ni universal, y no está disponible en todos los modelos o plataformas.

VI. CONSIDERACIONES PRÁCTICAS Y LIMITACIONES

5. GENERACIÓN DE ALUCINACIONES O CONTENIDO FALSO

Una de las limitaciones más significativas de los LLM es la tendencia a generar respuestas falsas, aunque plausibles. Este fenómeno, conocido como alucinación, ocurre cuando el modelo desconoce la respuesta correcta, pero produce una salida coherente y convincente desde el punto de vista lingüístico, aunque sea totalmente errónea.
Estas alucinaciones pueden tomar la forma de datos inventados, referencias inexistentes y argumentos incorrectos formulados con gran fluidez.

Aunque algunos modelos avanzados han mejorado en la detección de incertidumbre, indicando cuándo no disponen de información suficiente, la alucinación sigue siendo un riesgo importante. Por ello, es esencial verificar las respuestas con fuentes externas, utilizar modelos conectados a buscadores o bases de datos confiables y promover un uso crítico y reflexivo, especialmente en contextos profesionales o académicos.

VI. CONSIDERACIONES PRÁCTICAS Y LIMITACIONES

CONTROL DE LA VERACIDAD EN LLM

Cuando buscamos reforzar la veracidad de las respuestas de un modelo de lenguaje grande (LLM), especialmente si se han detectado errores previos, es posible instruirlo para que no genere información si carece de datos suficientes. Se le puede indicar explícitamente: "Si no sabes la respuesta, no inventes información; aclara que no puedes responder por falta de datos", o bien, "Si no sabes la respuesta, no inventes información; pídeme más contexto".

Limitaciones del Control Declarativo

A pesar de estas instrucciones, los LLM no poseen una capacidad innata para comprender y seguir directivas específicas como "no generar si no hay datos". El modelo puede tanto ignorar la instrucción y generar una respuesta con la información disponible, como reconocerla y abstenerse de responder. Esta posibilidad de que los LLM omitan la instrucción de no inventar subraya los límites del control declarativo sobre su comportamiento. La naturaleza probabilística de los LLM puede llevarlos a priorizar una respuesta plausible sobre el cumplimiento estricto de una instrucción de "no sé". Esto recalca la necesidad continua de verificación externa y análisis crítico de las respuestas generadas por la IA.

BIBLIOGRAFÍA

01 Beyond Prompt Content: Enhancing LLM Performance via Content-Format Integrated Prompt Optimization- arXiv, https://arxiv.org/html/2502.04295v3

02 A guide to prompt design: foundations and applications for healthcare simulation- Frontiers, https://www.frontiersin.org/journals/medicine/articles/10.3389/fmed.2024.1504532/full

03 Efficient Prompting Methods for Large Language Models: A Survey - arXiv, https://arxiv.org/html/2404.01077v2

04 Prompt engineering techniques: Top 5 for 2025 - K2view, https://www.k2view.com/blog/prompt-engineering-techniques/

05 What is few shot prompting? - IBM, https://www.ibm.com/think/topics/few-shot-prompting

BIBLIOGRAFÍA

06 What is chain of thought (CoT) prompting? - IBM, https://www.ibm.com/think/topics/chain-of-thoughts

07 A Survey of Frontiers in LLM Reasoning: Inference Scaling, Learning to Reason, and Agentic Systems - arXiv, https://arxiv.org/html/2504.09037v1

08 The Ultimate Guide to LLM Reasoning (2025) - Kili Technology, https://kili-technology.com/large-language-models-llms/llm-reasoning-guide

09 Reasoning Beyond Limits: Advances and Open Problems for LLMs - arXiv, https://arxiv.org/html/2503.22732v1

10 Inference-Time Compute Scaling Methods to Improve Reasoning Models, https://sebastianraschka.com/blog/2025/state-of-llm-reasoning-and-inference-scaling.html