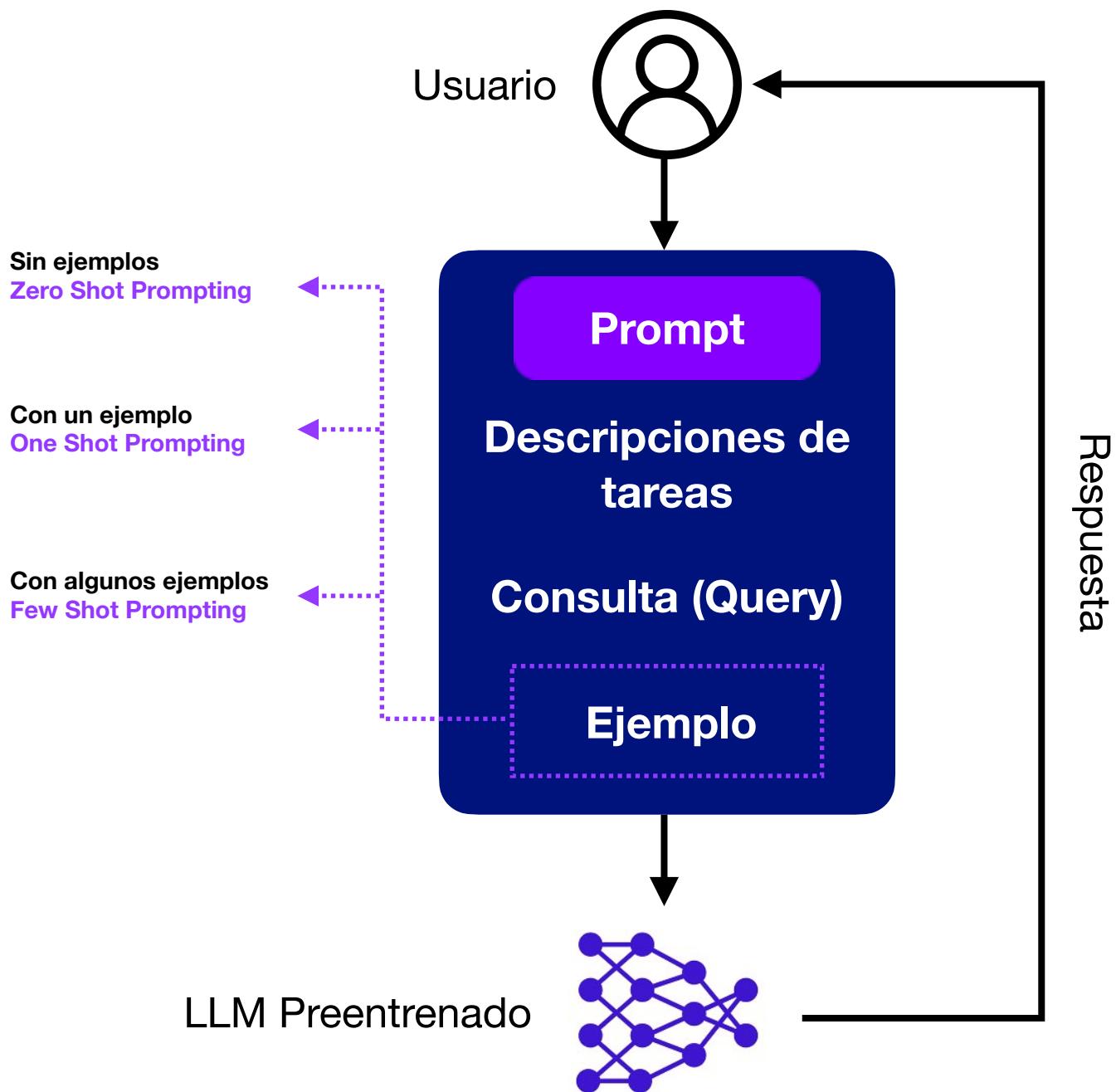


Guía completa de ingeniería de prompts



Por Antonio Richaud

¿Qué es un Prompt?

- Un **prompt** es una entrada que le das a un modelo de lenguaje, como parte del aprendizaje automático. Ya sea al trabajar con modelos como GPT, BERT, T5 u otros modelos avanzados, el prompt es la instrucción, consulta o contexto que orienta al modelo para generar una respuesta o completar una tarea.
- Los prompts pueden adoptar diferentes formas: una pregunta directa, una declaración a completar, o una serie de instrucciones detalladas. En cualquier caso, el objetivo principal es guiar al modelo para que entienda y resuelva una tarea de manera precisa y relevante.

Ejemplo general: Imagina que quieres que un modelo genere ideas para nombres de empresas; el prompt podría ser: “*Genera 10 ideas creativas para nombres de una nueva empresa de tecnología enfocada en IA.*”



Aspectos clave de un Prompt

- **Instrucción:** Un buen prompt debe tener instrucciones claras. Define exactamente lo que quieres que el modelo haga. Por ejemplo, “Resumir el contenido de este artículo” es una instrucción específica que orienta al modelo hacia una tarea de síntesis.
- **Contexto:** Proporcionar contexto es crucial para que el modelo genere respuestas más relevantes y coherentes. El contexto puede incluir detalles sobre el tono deseado, información sobre la audiencia objetivo, o ejemplos previos de la tarea. Esto ayuda a modelos como GPT-3, BERT, LLaMA, o modelos de código abierto como BLOOM, a entender mejor el propósito de la tarea.
- **Claridad y especificidad:** Asegúrate de que el prompt sea directo y sin ambigüedades. Cuanto más específico sea, mejores resultados obtendrás. Esto es especialmente importante cuando trabajas con modelos de lenguaje, ya que cualquier ambigüedad puede llevar a interpretaciones erróneas.

Consejo práctico: Si tu prompt es demasiado general o vago, el modelo puede tener dificultades para entender qué es lo que realmente necesitas. Usa palabras claves específicas para evitar respuestas abiertas o fuera de contexto.

Ejemplos de Prompts efectivos

- **Pregunta directa:** “¿Cuáles son las diferencias clave entre Python y Java para desarrollo web?”
- **Instrucción de tarea:** “Traduce la siguiente oración al francés y alemán: ‘La inteligencia artificial cambiará el futuro de la tecnología.’”
- **Descripción creativa:** “Imagina un mundo donde la energía solar es la única fuente de electricidad. Escribe una historia corta explorando cómo cambiaría la vida diaria de las personas.”
- **Análisis técnico:** “Analiza y comenta el siguiente fragmento de código en Python, destacando sus puntos fuertes y áreas de mejora.”

Este enfoque no solo mejora la claridad y relevancia del prompt, sino que puede ser aplicado a diversos modelos de lenguaje para obtener respuestas óptimas. Además, recuerda que la forma en que estructuras tu prompt puede influir enormemente en la calidad de la respuesta generada por el modelo.

Ingeniería de Prompts

La Ingeniería de Prompts es el arte de diseñar, ajustar y refinar las instrucciones o “prompts” con el objetivo de obtener los mejores resultados posibles de un modelo de lenguaje. No se trata solo de escribir una consulta al azar; se requiere una cuidadosa planificación y un conocimiento profundo de cómo los modelos interpretan y responden a los diferentes tipos de prompts.

- Este proceso implica reescribir y ajustar continuamente los prompts para optimizar la calidad y relevancia de las respuestas generadas por el modelo. A través de iteraciones sucesivas, puedes afinar los resultados y orientar al modelo para cumplir con objetivos específicos.
- La ingeniería de Prompts es una combinación dinámica de **creatividad** y **estrategia**. Implica experimentar con diferentes formulaciones, probar cómo responde el modelo, y luego refinar esos prompts para obtener respuestas cada vez más precisas y útiles. Esto es clave para modelos como GPT, BERT, LLaMA, BLOOM, y muchos otros modelos de lenguaje natural o de tareas específicas.

Sugerencia avanzada: Al trabajar con modelos multilingües, es importante recordar que un prompt en un idioma específico puede generar resultados diferentes en comparación con el mismo prompt en otro idioma, debido a las particularidades de interpretación lingüística del modelo.

Ventajas de la ingeniería de Prompts

- **No requiere entrenamiento adicional:** Lograr los resultados deseados no depende de reentrenar el modelo desde cero, lo que ahorra tiempo y recursos computacionales.
- **Eficiente en costo y tiempo:** La creación y ajuste de prompts es un proceso rápido que permite obtener resultados inmediatos. Esto resulta mucho más eficiente que la reconfiguración o el ajuste fino de los pesos internos del modelo.
- **Flexibilidad de tareas:** Los prompts pueden ser adaptados a una gran variedad de tareas, desde traducción y resumen de texto hasta generación de código y respuestas a preguntas de conocimiento general.
- **Baja barrera de entrada:** Incluso usuarios con conocimientos técnicos limitados pueden crear y utilizar prompts efectivos con solo aprender algunas reglas básicas de formulación.
- **Prototipado rápido:** Permite iterar y probar diferentes prompts en un corto periodo de tiempo para descubrir cuál produce las respuestas más adecuadas a tus necesidades. Esta capacidad de iterar rápidamente es especialmente útil en el desarrollo de chatbots, asistentes virtuales y otras aplicaciones conversacionales.
- **Escalable:** La misma técnica puede ser usada para trabajar en múltiples dominios, desde servicio al cliente hasta análisis financiero o educativo, simplemente ajustando los prompts para cada contexto.

- **Aprovecha modelos preentrenados:** La Ingeniería de Prompts hace uso de los conocimientos ya existentes en el modelo. Los modelos de lenguaje preentrenados tienen vastos conocimientos y habilidades, y un buen prompt puede aprovechar toda esa información sin la necesidad de entrenar de nuevo.

Nota adicional: Si bien la Ingeniería de Prompts puede ofrecer resultados impresionantes, es importante tener en cuenta que, en ciertos casos, un prompt mal formulado puede llevar a respuestas poco claras o irrelevantes. Por ello, se recomienda siempre probar múltiples variaciones y comparar resultados.

Desventajas de la ingeniería de Prompts

- **Resultados inconsistentes:** Incluso pequeños cambios en el prompt pueden conducir a resultados impredecibles o inesperados. Esto se debe a que los modelos de lenguaje, al ser sensibles a la formulación exacta de las palabras, pueden interpretar de manera diferente cada variación.

Consejo para mitigar: Realiza pruebas con varias versiones de un mismo prompt para identificar cuál genera respuestas más coherentes y útiles.

- **Método de prueba y error:** Se requiere experimentar con muchas iteraciones para encontrar el prompt óptimo. Esto puede consumir tiempo, especialmente en tareas complejas donde pequeños ajustes pueden llevar a grandes diferencias en la calidad de las respuestas.

Enfoque proactivo: Usa herramientas de automatización o frameworks especializados para probar rápidamente múltiples variaciones de prompts y evaluar sus resultados.

- **Control limitado:** A pesar de que se puede guiar al modelo con prompts, existe una falta de control fino sobre el comportamiento del modelo. No siempre es posible garantizar que el modelo responderá exactamente como se espera, lo que puede resultar en respuestas no deseadas o irrelevantes.

Solución alternativa: Si necesitas más control sobre el modelo, podría ser mejor entrenar un modelo específico para tu tarea en lugar de depender únicamente de prompts en un modelo preentrenado.

- **Dificultad con Prompts complejos:** A medida que las tareas se vuelven más específicas o complejas, diseñar prompts efectivos se convierte en un reto. Por ejemplo, si se espera que el modelo realice tareas que requieren razonamiento avanzado o múltiples pasos, un solo prompt podría no ser suficiente.

Recomendación: Divide la tarea en prompts más pequeños y manejables, y evalúa si puedes construir una “conversación” progresiva con el modelo para obtener la respuesta deseada.

- **Dependencia del preentrenamiento:** Los resultados obtenidos están limitados por el conocimiento y los sesgos ya presentes en el modelo preentrenado. Si el modelo no ha sido entrenado en un dominio específico, su capacidad para generar respuestas precisas será limitada.

Possible solución: Considera utilizar modelos especializados o actualizados en lugar de modelos generales, especialmente si trabajas en un dominio técnico o nicho.

- **Problemas de escalabilidad:** La Ingeniería de Prompts puede no escalar bien para aplicaciones altamente especializadas o para casos en los que se requieren respuestas extremadamente precisas. En aplicaciones donde se necesite mantener un alto grado de consistencia y exactitud, los prompts pueden quedarse cortos.

Alternativa escalable: Para tareas muy específicas, puede ser más efectivo desarrollar una arquitectura híbrida que combine prompts con otros enfoques, como ajuste fino del modelo (fine-tuning) o modelos complementarios que actúen como filtros adicionales para mejorar la calidad de la respuesta.

Técnicas de ingeniería de Prompts

Few-shot prompting

Consiste en proporcionar al modelo ejemplos específicos (normalmente de 2 a 5) de la tarea que quieras realizar. Esto le ayuda a entender el tipo de respuesta que deseas.

Alcances: Los ejemplos proporcionan un contexto fuerte para el modelo, guiándolo a seguir un patrón preestablecido en sus respuestas. Esto es útil para tareas complejas o específicas como traducción, corrección de texto, o generación de código.

Limitaciones: La efectividad de los ejemplos depende de la calidad y relevancia de los mismos. Si los ejemplos no están bien elegidos, el modelo puede producir resultados inconsistentes. Además, proporcionar demasiados ejemplos puede limitar la creatividad del modelo o hacer que responda con menos diversidad.

Técnica ASPECT

ASPECT es un acrónimo que ayuda a diseñar prompts efectivos. La idea es asegurarse de que el prompt sea claro, conciso y específico. Aquí te dejo el desglose:

- **A (Action)**: Especifica la acción que el modelo debe realizar. Por ejemplo, “Explica”, “Resume”, “Genera”, etc.
- **S (Subject)**: Define el tema o contexto del prompt. Esto ayuda a centrar la respuesta en el tema específico que necesitas.
- **P (Purpose)**: Indica el propósito o la intención detrás del prompt. Por ejemplo, si quieras generar ideas creativas, opiniones, o explicar un concepto complejo.
- **E (Examples)**: Proporciona ejemplos que el modelo puede seguir. Esto puede ser especialmente útil para guiar el formato o el tono de la respuesta.
- **C (Constraints)**: Especifica cualquier restricción que debas tener en cuenta, como la longitud de la respuesta, el idioma, o el estilo.
- **T (Tone)**: Define el tono deseado, como formal, técnico, conversacional, humorístico, etc.

Alcances: Esta técnica ayuda a diseñar prompts que sean más detallados y enfocados, lo que aumenta la probabilidad de obtener respuestas relevantes y útiles. Es especialmente efectiva cuando buscas respuestas específicas en lugar de generalidades.

Limitaciones: El proceso de construir prompts tan detallados puede consumir más tiempo, y no siempre garantiza que el modelo siga todas las instrucciones exactamente, especialmente si algunas de las restricciones son muy complejas.

Chain-of-thought prompting

Es una técnica que anima al modelo a “pensar en voz alta”, proporcionando razonamiento paso a paso antes de dar la respuesta final. Es especialmente útil para problemas que requieren lógica o análisis complejo.

Alcances: Mejora la capacidad del modelo para resolver tareas que requieren múltiples pasos o razonamiento lógico. Ayuda a reducir respuestas incorrectas en tareas como matemáticas, planificación o cualquier problema que requiera secuencia de ideas.

Limitaciones: Si el prompt no guía adecuadamente al modelo en el proceso de pensamiento, puede llevar a pasos de razonamiento incorrectos o irrelevantes. Además, puede resultar en respuestas largas y redundantes si no se estructura de manera adecuada.

Prompt layering (Prompt en capas)

Esta técnica implica dividir una tarea compleja en varios prompts más pequeños y secuenciales. Cada prompt se utiliza para obtener una parte específica de la información, que luego se combina para formar la respuesta final.

Alcances: Ayuda a mantener un control más preciso sobre la generación de la respuesta. Es útil para obtener detalles granulares sobre diferentes aspectos de una misma pregunta o para descomponer un problema en sus componentes esenciales.

Limitaciones: Este enfoque puede aumentar la complejidad del proceso y la duración de la interacción con el modelo. Además, puede requerir una planificación cuidadosa para asegurar que cada capa esté correctamente alineada con el resultado final deseado.

Instructive prompting

En esta técnica, proporcionas al modelo instrucciones claras y específicas sobre cómo se espera que responda. Esto incluye el formato, la estructura, el tono, o incluso qué tipo de información incluir o evitar.

Alcances: Útil cuando se requiere que el modelo siga estrictamente ciertas reglas de formato o tono. Por ejemplo, “Responde con una lista de puntos clave”, o “Da una respuesta concisa de no más de 100 palabras”.

Limitaciones: Al limitar al modelo a instrucciones estrictas, puede perderse información valiosa que podría haber generado de manera más libre. Además, algunos modelos pueden interpretar las instrucciones de manera diferente dependiendo de la formulación exacta.

Iterative prompt refinement (Refinamiento iterativo)

Es la práctica de mejorar un prompt con base en las respuestas obtenidas. Si una respuesta no es lo suficientemente relevante, el prompt se ajusta y se vuelve a intentar, permitiendo que el proceso se refine con cada iteración.

Alcances: Es una técnica que permite obtener resultados óptimos a través de un proceso continuo de prueba y ajuste. Útil para casos en los que se necesita un alto nivel de precisión o detalle en las respuestas.

Limitaciones: Este enfoque requiere tiempo y paciencia para iterar sobre los prompts y evaluar los resultados. Además, a veces puede ser difícil identificar exactamente qué parte del prompt necesita ajustes.

Estas técnicas te ayudarán a crear prompts más efectivos y precisos para cualquier modelo de lenguaje. Además, es importante recordar que experimentar con diferentes enfoques y combinaciones de estas técnicas puede llevar a obtener mejores resultados. Mucho éxito comunicándote de la mejor manera con tu LLM de confianza. :)