Prompt Engieneering

Introducción

Alipio Laboriano Galindo

Pontificia Universidad Católica del Perú

February 1, 2025

Contenido

Prompt Engineering

Prompt

- Un "prompt" es lo que el usuario le pide a un modelo de AI que haga.
- Este pedido se hace mediante una entrada textual, que puede ser tan simple como una pregunta o tan compleja como un conjunto de datos con contexto adicional y hasta ejemplos.
- El prompt guía al modelo para producir los resultados deseados por el usuiario.

Ejemplo:
prompt = *"""Continúa la siguiente historia. Escribelo en menos de 50 palabras. Érase una vez, en un mundo donde los animales podían hablar, un valiente ratón llamado Benjamín decidió..."""*

Prompt Engineering

Prompting

- El "prompting" es el proceso o la forma en que los humanos se comunican con la AI.
- Consiste en utilizar un lenguaje humano adaptado para decirle a la AI qué queremos y cómo lo queremos.
- Es el acto de interactuar con la AI a través de prompts para obtener resultados específicos.

Prompt Engineering

El Prompt Engineering es una disciplina que se enfoca en crear y optimizar prompts de manera efectiva para aprovechar al máximo los modelos de lenguaje generativo. Este campo permite traducir ideas de un lenguaje conversacional común a instrucciones más precisas y optimizadas, maximizando el rendimiento de los modelos de AI.

¿Por qué es importante el Promting?

- Guía al modelo de AI, para generar los resultados más relevantes, los cuales son coherentes con el contexto dado y devuelven los resultados en el formato especificado.
- Aumenta el control e interpretabilidad y reduce los sesgos potenciales que puedan existir.
- Diferentes modelos, responden de manera diferente al mismo prompt, por lo que es necesario conocer los detalles de cada modelo, para diseñar un buen promt.
- Reduce el tema de halucinaciones presentes en todos los LLM's.

Optimización de Prompts

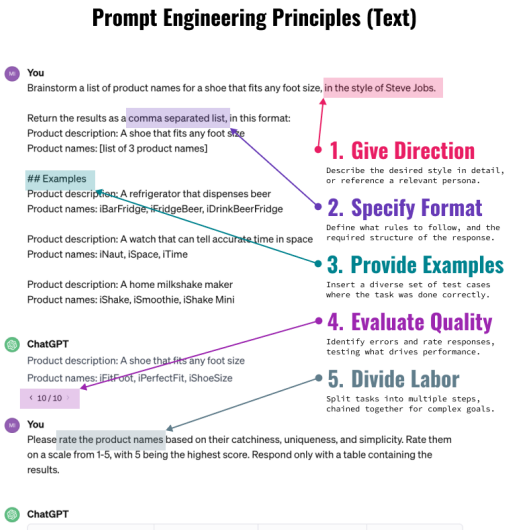
Los LLMs ofrecen un poder inmenso para automatizar diversas tareas, pero su efectividad depende de la calidad de los prompts que se les proporcione para que estos devuelvan los resultados que el usuario espera, por lo cual un **prompt bien diseñado y optimizado**, devolverá información de mayor calidad y más precisa.

Optimización de Prompts

Los 5 principios del prompting [?]

- **Dar dirección**. Consiste en describir el estilo deseado en detalle o hacer una referencia a una persona relevante.
- **Especificar formato**.Consiste en definir qué reglas se debe seguir y la estructura requerida de la respuesta.
- **Proveer ejemplos**. Consiste en dar un conjunto diverso de casos de prueba donde la tarea se haya realizado correctamente.
- **Evaluar calidad**. Consiste en identificar errores y calificar las respuestas, midiendo el rendimiento o eficacia con la que el modelo realiza, la tarea asignada.
- **Dividir el trabajo**. Consiste en dividir las tareas en múltiples pasos, para alcanzar objetivos complejos.

Los 5 principios del prompting [?]



Optimización de Prompts

Elementos de un Prompt [?]

- **Instrucción**. Una tarea específica o instrucción que deseas que el modelo realice.
- **Contexto**.Información externa o contexto adicional que puede guiar al modelo hacia mejores respuestas.
- **Inputs**. La entrada o pregunta para la cual estamos interesados en obtener una respuesta.
- **Outputs**. El tipo o formato del resultado esperado
- **Dividir el trabajo**. Consiste en dividir las tareas en múltiples pasos, para alcanzar objetivos complejos.

Prompt Engineering Guide.

Optimización de Prompts

Buenas prácticas [?]

ChatGPT, considera las siguientes 6 estrategías, para conseguir mejores resultados.

- **Escribe instrucciones claras**. Incluir detalles, darle un rol al modelo, etc.
- **Proporcionar texto de referencia**. Agregar contexto.
- **Dividir tareas complejas en subtareas**. Las tareas complejas tienden a tener tasas de error más altas que las tareas más simples.
- **Tomar un tiempo para "pensar"**.Los modelos cometen más errores cuando intentan responder de inmediato, en vez de tomarse un tiempo para dar su respuesta.
- **Utilice herramientas externas**. Compense las debilidades del modelo alimentándolo con información adicional (Sistemas RAG).
- **Probar los cambios sistemáticamente**.Es más fácil mejorar el rendimiento si se puede medir.

Enhance results with prompt engineering strategies.

Optimización de Prompts

Tips for Effective Prompt Engineering [?]

- **Ser específico**. Incluye suficiente contexto y detalle para guiar a la AI hacia el resultado deseado.
- **Ser Consiso**. Asegúrate de que el prompt sea directo y claro, eliminando ambigüedades.
- **Solicita información detallada**. Pide al modelo que describa por qué toma ciertas decisiones, lo cual puede ayudar a generar soluciones más precisas, especialmente en tareas complejas
- **Darle un rol**.Define explícitamente el rol del modelo (por ejemplo, como un "coach de vida") para que sus respuestas se alineen con el propósito esperado.
- **Darle Ejemplos**. Incluye ejemplos que demuestren el tono, estilo y tipo de respuesta esperada, ayudando al modelo a comprender cómo debe responder.
- **Distingue entre los ejemplos y la consulta real**.Separa claramente los ejemplos de la pregunta del usuario.

Optimización de Prompts

Template

- **Rol - Role**. Asignarle un rol o una especializacización al modelo.
- **Tarea - Task**. Decirle al modelo lo que queremos que haga. Incluso se puede detallar el paso a paso.
- **Contexto - Context**. Darle al modelo información adicional, definición de conceptos, fuentes, etc.
- **Darle Ejemplos**. Incluye ejemplos que demuestren el tono, estilo y tipo de respuesta esperada, ayudando al modelo a comprender cómo debe responder.
- **Salidas - Outputs**.Darle al modelo la estructura de la salida deseada, puede especificar el formato, la longitud, etc.
- **Evaluar la calidad de los resultados**.Verificar vs el grounth truth, pedir que el modelo devuelva el confidence score, medir el rendimiento en las tareas asignadas.

Prompting Techniques

A continuación se presenta diferentes técnicas más avanzadas de ingeniería de prompts que permiten realizar tareas más complejas y mejorar la fiabilidad y el rendimiento de los LLMs.

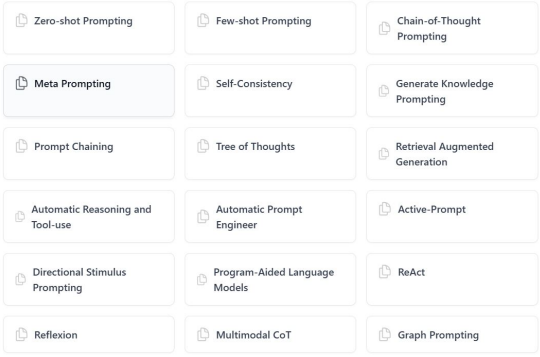


Figure: Prompting Techniques

Prompting Techniques

Zero-Shoot Prompting

- El zero-shot prompting consiste en pedir a un modelo que genere una respuesta sin proporcionar ningún tipo de ejemplo.
- Muchas tareas, ya están dentro de las capacidades de los LLMs, lo que les permite ofrecer excelentes respuestas incluso sin ejemplos o guías detalladas.

Prompting Techniques

Few-Shoot Prompting

- En esta técnica, el prompt generalmente incluye pocos ejemplos o entradas acompañadas de sus respectivas respuestas.
- El modelo de lenguaje aprende de estos ejemplos y aplica lo aprendido para responder preguntas similares.
- A diferencia del zero-shot prompting, donde el modelo genera respuestas para tareas completamente nuevas, el few-shot prompting aprovecha ejemplos en el prompt para mejorar su desempeño.

Prompting Techniques

Role Prompting

- El role prompting consiste en instruir al modelo de lenguaje (LLM) para que asuma un rol o identidad específica al ejecutar una tarea, como actuar como un redactor publicitario (copywriter).
- Esta instrucción puede influir en la respuesta del modelo al proporcionar un contexto o perspectiva para la tarea.
- Cuando se trabaja con role prompts, el proceso iterativo incluye: 1) Definir el rol en el prompt, 2) Utilizar el prompt para generar una respuesta del LLM y 3) Evaluar la respuesta y refinar el prompt

Prompting Techniques

Chain Prompting

- El Chain Prompting consiste en enlazar una serie de prompts de manera secuencial, donde la salida de un prompt sirve como entrada para el siguiente.
- Al implementar Chain Prompting (con LangChain), se deben considerar los siguientes pasos:
  - Identificar y extraer información relevante de la respuesta generada.
  - Desarrollar un nuevo prompt utilizando la información extraída, asegurándose de que se base en la respuesta anterior.
  - Continuar este proceso tantas veces como sea necesario hasta alcanzar el resultado deseado.

Chain of Thought Prompting

- El Chain of Thought Prompting (CoT) es una técnica que permite a los modelos de lenguaje expresar su proceso de razonamiento, mejorando la precisión de sus respuestas.
- Consiste en proporcionar ejemplos que demuestren el proceso lógico, guiando al modelo para que explique su razonamiento mientras responde. Esta estrategia ha demostrado ser útil en tareas de aritmética, razonamiento lógico y pensamiento simbólico.

Retrieval Augmented Generation (RAG)

- La Recuperación de Generación Aumentada (RAG) permite a los modelos de lenguaje acceder a fuentes externas de información para tareas complejas, mejorando la coherencia fáctica (la información proporcionada se alinea con datos o conocimientos verdaderos y verificables) y reduciendo errores como las "alucinaciones".
- Esto es crucial cuando los datos evolucionan con el tiempo, ya que evita reentrenar el modelo completo y permite acceder a información reciente. RAG resuelve la limitación del conocimiento estático de los modelos de lenguaje.

GRACIAS