Automatización de Respuestas de Servicio al Cliente con Prompt Engineering

# Portada

Nombre y apellido del autor: Gino Zampierón  
Nombre del curso: Prompt Engineering  
Nª de comisión: 71380

# Introducción

En esta segunda entrega, profundizaremos en el Proyecto Final titulado "Automatización de Respuestas de Servicio al Cliente con Prompt Engineering". El objetivo es desarrollar una prueba de concepto (POC) que demuestre la efectividad de los prompts generados para automatizar respuestas de servicio al cliente, mejorando su calidad y personalización.

# Presentación del Problema a Abordar

El problema elegido es la automatización de respuestas de servicio al cliente utilizando prompts de IA. Las empresas reciben numerosas consultas de clientes a diario, muchas de las cuales son repetitivas y pueden ser respondidas automáticamente. Sin embargo, las respuestas automáticas actuales a menudo carecen de personalización y contexto, lo que lleva a una baja satisfacción del cliente.  
  
Por ejemplo, respuestas genéricas a consultas sobre estados de pedidos o políticas de devolución pueden frustrar a los clientes que buscan información específica sobre su caso. El problema consiste en mejorar la calidad y personalización de las respuestas automáticas mediante el uso de prompts optimizados que consideren el contexto de la consulta y el historial del cliente.

# Desarrollo de la Propuesta de Solución

La solución propuesta consiste en utilizar modelos de IA para generar respuestas personalizadas a consultas de clientes. Utilizaremos técnicas de prompting avanzadas (Zero, One, y Few Shot Prompting) para mejorar la relevancia y calidad de las respuestas automáticas.  
  
Implementación Técnica:  
El desarrollo se llevará a cabo en un Jupyter Notebook, utilizando modelos de texto-texto y texto-imagen. A continuación se presentan los bloques de código que se utilizarán:

## Código de Implementación (Texto-Texto)

```python  
from transformers import pipeline  
  
# Ejemplo de Zero Shot Prompting  
generator = pipeline('text-generation', model='gpt-3.5-turbo')  
prompt\_zero\_shot = "Genera una respuesta para un cliente que pregunta sobre el estado de su pedido."  
respuesta\_zero\_shot = generator(prompt\_zero\_shot, max\_length=50)  
print(respuesta\_zero\_shot)  
```

## Código de Implementación (Texto-Imagen)

```python  
from dalle\_pytorch import DALLE  
dalle = DALLE.load\_model("path\_to\_model")  
  
prompt\_texto\_imagen = "Genera una imagen que represente la entrega de un pedido en tiempo récord."  
imagen = dalle.generate\_images(prompt\_texto\_imagen)  
display(imagen)  
```

\*\*Insertar ilustraciones aquí:\*\* Incluye capturas de pantalla del código de Jupyter Notebook en ejecución, mostrando los resultados generados tanto por el modelo de texto-texto como por el modelo de texto-imagen. Estas ilustraciones deben ubicarse después de cada bloque de código correspondiente.

# Justificación de la Viabilidad del Proyecto

La viabilidad técnica de este proyecto es alta, dado que se utilizan modelos de IA preentrenados y herramientas disponibles como GPT-3 y DALL-E. Se cuenta con los recursos computacionales necesarios para ejecutar los modelos y obtener resultados en tiempo real.  
  
Para asegurar el éxito del proyecto, se implementará en las siguientes fases:  
- Mes 1: Investigación y recopilación de datos.  
- Mes 2: Desarrollo de prompts iniciales y pruebas.  
- Mes 3: Obtención de feedback y optimización inicial.  
- Mes 4: Integración técnica.  
- Mes 5-6: Evaluación, ajustes finales, y mejora continua.

# Objetivos

- Demostrar la comprensión de los principios y técnicas detrás de Fast Prompting.  
- Experimentar con diferentes configuraciones de prompts para optimizar la eficacia.  
- Preparar una demostración efectiva en el Jupyter Notebook para mostrar el funcionamiento de la POC.  
- Analizar si las nuevas técnicas aprendidas permiten mejorar la propuesta de solución planteada en la Preentrega 1.

# Metodología

El proyecto se llevará a cabo en varias etapas, comenzando con la investigación inicial, seguido del desarrollo y prueba de prompts, y terminando con la integración y evaluación final. Se utilizarán técnicas de Zero, One y Few Shot Prompting para optimizar las respuestas automáticas y mejorar la satisfacción del cliente.

# Herramientas y Tecnologías

- GPT-3 para generación de texto.  
- DALL-E para generación de imágenes.  
- Jupyter Notebook para desarrollo y presentación del código.

# Implementación

Desarrolla y presenta el código con ejemplos claros y comentarios. Incluye los siguientes bloques de código en tu Jupyter Notebook: