

Optimización de la organización y planificación de tareas mediante Inteligencia Artificial

Autor: Joaquín Ezequiel Bujakiewicz

Curso: Inteligencia Artificial: Generación de Prompts

Comisión: 86420

Proyecto: Optimización de la organización y planificación de tareas mediante Inteligencia Artificial.

Problema a abordar

En contextos laborales y académicos, muchas personas presentan dificultades para organizar tareas, priorizar actividades y estructurar información de manera clara. Esto genera pérdida de tiempo, desorden en los procesos de trabajo y una disminución de la productividad. La problemática se vuelve más relevante cuando se manejan múltiples responsabilidades o grandes volúmenes de información sin una metodología de organización definida, lo que dificulta la toma de decisiones y el cumplimiento de objetivos.

Propuesta de solución

La propuesta consiste en desarrollar un sistema de generación de prompts que utilice modelos de IA de **texto a texto** y **texto a imagen** para resolver problemas de organización y planificación de tareas.

Mediante prompts de texto a texto, se transformará información desordenada en listas de tareas estructuradas, planes de acción y cronogramas claros. Posteriormente, a través de prompts de texto a imagen, se generarán representaciones visuales como diagramas de flujo y esquemas organizativos que permitan visualizar de forma simple la distribución y secuencia de las tareas, facilitando su comprensión y seguimiento.

Los prompts estarán orientados a:

- Organizar información en tareas priorizadas.
 - Generar planes de acción paso a paso.
 - Crear imágenes que representen visualmente la planificación y el flujo de trabajo.
-

Metodología

El proyecto se desarrollará mediante las siguientes etapas:

Definición del problema y análisis de necesidades.

Diseño estratégico de prompts según complejidad.

Implementación en GoogleCollab

Evaluación de eficiencia (cantidad de consultas y consumo de tokens).

Ajuste iterativo para optimización.

Herramientas y Tecnologías:

Python

GoogleCollab

Modelos de lenguaje (texto → texto)

Herramientas de generación de imagen (texto → imagen)

Técnicas de Fast Prompting (zero-shot, one-shot, few-shot)

Desarrollo

El sistema trabajara en dos etapas:

Texto → Texto: Organización, priorización y estructuración de tareas.

Texto → Imagen: Generación de una representación visual del plan de acción.

Se aplicarán técnicas de prompting según la complejidad de cada etapa.

Técnicas de Prompting Utilizadas

Zero-Shot Prompting

Se utiliza zero-shot prompting en la etapa de organización inicial de tareas, el modelo recibirá una instrucción sin ejemplos previos para priorizar y estructurar tareas según urgencia. La tarea de organización es comprendida por los modelos de lenguaje actuales. No requiere ejemplos adicionales para obtener resultados.

El uso de zero-shot permite:

- Reducir la cantidad de tokens.
- Disminuir el costo de cada consulta.
- Mantener simplicidad.
- Optimizar la eficiencia.

Ofrece el mejor balance entre rendimiento y costo en tareas estándar.

One-Shot Prompting

Se aplica one-shot prompting cuando se necesita asegurar un formato específico de salida. El uso de un único ejemplo permite guiar al modelo para que mantenga consistencia sin aumentar significativamente el consumo de tokens.

El one-shot prompting:

- Reduce errores de formato.
- Disminuye la necesidad de reconsultas.
- Mejora la claridad de la salida.

Aporta control adicional sin el costo elevado del few-shot.

Few-Shot Prompting

Se utiliza few-shot prompting únicamente en tareas más complejas, como la transformación del plan en una estructura que sirva como base para la generación de imágenes.

El few-shot prompting mejora la coherencia cuando la tarea requiere mantener un patrón específico.

Su uso se limita a situaciones donde:

- La complejidad es mayor.

- Se requiere mayor precisión.
- El beneficio justifica el incremento en tokens.

Se evita su aplicación innecesaria para mantener la rentabilidad del proyecto.

Optimización y eficiencia

El proyecto está diseñado para minimizar consultas a la API y reducir costos operativos.

- Se utiliza zero-shot cuando la tarea lo permite.
- Se incorpora one-shot solo para controlar el formato.
- Se limita el few-shot a casos donde mejora significativamente la calidad.

De esta manera, el sistema demuestra una aplicación estratégica de técnicas de Fast Prompting, priorizando eficiencia, y viabilidad económica.

En el modelo presentado en github utilice código para hacer 1 prompt con cada estilo y realizar una comparación de tokens consumidos para luego elegir el más eficiente y usar el resultado más eficiente para la generación de imagen, en el modelo final va a usar solo 1 prompt textual y 1 de imagen.