Informe breve

Estudiantes:

- Leidis Lopez
- Camilo Perez
- Santago Muñoz

Repositorio de GitHub: https://github.com/ceperezm/App_traductor_gradio-mlflow_docker.git

Arquitectura y uso de *App_traductor_gradio-mlflow_docker*

Arquitectura
La aplicación está compuesta por **dos contenedores Docker**:

1. **MLflow Server**

   Imagen usada: ghcr.io/mlflow/mlflow:v2.10.0 según el README.

   Puerto: **5000** para acceder a la interfaz web de MLflow.

   Volúmenes montados desde el host:

   mlruns/ para los experimentos (--backend-store-uri file:///mlflow/mlruns)

   mlartifacts/ para artefactos (--default-artifact-root file:///mlflow/mlartifacts)

   Red: conectado a una red Docker llamada traductor-network, para que el contenedor del traductor pueda comunicarse con él.

Aplicación Gradio (traductor-app)

Imagen personalizada ceperezm/traductor-genai:1.0.0

Puerto: **7860**,puerto por defecto de Gradio.

Variables de entorno: se pasa la API key usando -e OPEN_ROUTER_API_KEY="tu-api-key-aqui" para que la app use el SDK de OpenRouter.

Volúmenes: también monta mlruns/ y mlartifacts/ del host dentro del contenedor para que la aplicación pueda registrar en MLflow los eventos y artefactos.

Red: mismo traductor-network para poder hablar con el servidor de MLflow.

Cómo pasar la API key

Se define la variable de entorno **OPEN_ROUTER_API_KEY** al ejecutar el contenedor de la aplicación Gradio.

```
docker run -d \
  --name traductor-app \
  --network traductor-network \
  -p 7860:7860 \
  -v $(pwd)/mlruns:/app/mlruns \
  -v $(pwd)/mlartifacts:/app/mlartifacts \
  -e OPEN_ROUTER_API_KEY="tu-api-key-aqui" \
  ceperezm/traductor-genai:1.0.0
```

Debes reemplazar "tu-api-key-aqui" con tu clave real.

Observaciones sobre latencia y calidad de traducción

Las mediciones muestran una latencia que oscila entre aproximadamente 3.8 y 8.4 segundos por traducción, dependiendo del modelo y de la longitud del texto. Este tiempo es aceptable para un uso general, aunque no adecuado para escenarios que requieren respuestas inmediatas. Los modelos más grandes incrementan de manera notable el tiempo de respuesta, y la latencia total incluye tanto la llamada a la API como el procesamiento del modelo.

En cuanto a la calidad, los valores de *length_ratio* entre 0.67 y 0.92 indican que las traducciones conservan una extensión proporcional al texto original y mantienen coherencia y fidelidad semántica. Los modelos utilizados generan resultados fluidos y naturales, comparables a los de sistemas avanzados de traducción automática. En general, la calidad es alta, con variaciones mínimas según la elección del modelo.