Explicación de las funciones de control de tokens

Hemos implementado funciones de control de tokens en nuestro código para asegurar que el número total de tokens (entrada más salida) no exceda el límite permitido por el modelo `gpt2`. Aquí está la explicación de las funciones clave que hemos creado y cómo puedes ajustarlas cuando cambies de modelo.

Función `truncate_context`

Esta función se utiliza para recortar el contexto de entrada de modo que el número total de tokens no exceda el límite permitido. Utilizamos el `GPT2Tokenizer` para contar y recortar los tokens.

```
def truncate_context(context, max_tokens):
    tokens = tokenizer.encode(context)
    if len(tokens) > max_tokens:
        tokens = tokens[:max_tokens]
        return tokenizer.decode(tokens, clean_up_tokenization_spaces=True)
    return context
```

Parámetros:

- `context`: El texto de contexto que se desea truncar.
- `max_tokens`: El número máximo de tokens permitidos para el contexto.

Descripción:

- La función tokeniza el contexto usando `tokenizer.encode`.
- Si el número de tokens excede `max_tokens`, se truncan los tokens al límite especificado.
- La función devuelve el contexto truncado como texto utilizando `tokenizer.decode`.

Definición de límites de tokens

Hemos definido los límites de tokens de entrada (`MAX_INPUT_TOKENS`) y salida (`MAX_NEW_TOKENS`) para asegurarnos de que el número total de tokens no exceda 1024, el límite para `gpt2`.

```
MAX_INPUT_TOKENS = 824  # Ajustar según sea necesario
MAX_NEW_TOKENS = 100
```

Descripción:

- `MAX_INPUT_TOKENS`: El número máximo de tokens permitidos para el contexto de entrada.
- `MAX_NEW_TOKENS`: El número máximo de tokens permitidos para la respuesta generada por el modelo.

Aplicación en el código

Utilizamos la función `truncate_context` para recortar el contexto antes de crear el prompt y enviar la solicitud al modelo.

```
# Obtener el contexto
context = " ".join([doc.page_content for doc in texts])
context = truncate_context(context, MAX_INPUT_TOKENS)

# Crear el prompt con el contexto recortado
prompt = prompt_template.format(context=context, question=question)
```

Ajustes para cambiar de modelo

Cuando se cambie de modelo, necesitaremos ajustar los límites de tokens (`MAX_INPUT_TOKENS` y `MAX_NEW_TOKENS`) según las especificaciones del nuevo modelo. La mayoría de los modelos tienen documentación que indica el límite máximo de tokens que pueden manejar. Por ejemplo, si se cambia a un modelo con un límite de 2048 tokens, hay que ajustar los límites de la siguiente manera:

```
MAX_INPUT_TOKENS = 1948  # Ajustar según el nuevo límite
MAX_NEW_TOKENS = 100
```

Asegurarse de revisar la documentación del nuevo modelo para conocer el límite exacto de tokens y ajustar las variables en consecuencia.

Resumen

- **truncate_context**: Recorta el contexto para no exceder el límite de tokens.
- **MAX_INPUT_TOKENS y MAX_NEW_TOKENS**: Define los límites de tokens de entrada y salida.
- **Ajustes para nuevos modelos**: Cambia los valores de `MAX_INPUT_TOKENS` y `MAX_NEW_TOKENS` según las especificaciones del nuevo modelo.

Estas funciones y ajustes aseguran que tu código maneje correctamente los límites de tokens, evitando errores y asegurando que las solicitudes al modelo se procesen correctamente.