Aplicaciones LLM

La nueva API de OpenAI como alternativa a Langchain

Intro

- Uno de los motivos que hicieron popular a LangChain en sus inicios es que era visto como una alternativa más sencilla y plural que la API de OpenAI.

- Esta tendencia podría cambiar debido a tres factores:
  - Hasta ahora el modelo ChatGPT de OpenAI es de largo el más utilizado, por lo que en este momento el valor de LangChain como framework multi-modelo prácticamente no tiene sentido.

  - Con la apuesta por el nuevo lenguaje LCEL, LangChain se está haciendo más complejo.

  - En el DevDay de Noviembre de 2023, OpenAI ha lanzado una versión más sencilla y versátil de su API.

Precaución

- Que la API de OpenAI pueda hacer algo no significa que sea la mejor forma de hacerlo:
  - Utilizar las nuevas funcionalidades de OpenAI puede ser muy caro.
  - Utilizar las nuevas funcionalidades de OpenAI implica tener un menor grado de conocimiento y control sobre las settings, dado que la API de OpenAI es opaca en muchos sentidos.

Análisis de la nueva API de OpenAI en tres capítulos

- En este capítulo vamos a presentar los principales cambios que OpenAI introduce en su API desde el DevDay de Noviembre de 2023, así como su impacto para los desarrolladores de aplicaciones LLM.

- En el segundo capítulo haremos un breve repaso de la API de OpenAI.

- Y dedicaremos un tercer capítulo a analizar con mayor detalle la funcionalidad más interesante de la API de OpenAI: las funciones OpenAI.

Principales cambios introducidos en la API de OpenAI

- Context Window de 128.000 tokens (modelo GPT-4 Turbo)
- Multiple function calling.
- JSON mode.
- Reproducible outputs (seed parameter).
- Assistants.
- RAG Assistant.
- Vision.
- Text to Speech.

GPT-4 Turbo with 128k context

- RAG is not necessary for simple cases.
- You can now upload one full book (up to 300 pages) and make questions about it.
  - But be careful: this is a very expensive functionality.
- GPT-3.5 Turbo has 16k context now.

Multiple function calling

- Before it, you had to call one function at a time. This simplifies the process and saves time.
- Now you do not pass in functions, but tools. The type of the tool is function.

JSON mode

- Before it, chatGPT responded with text (strings) and we had to use an Output Parser to convert it to JSON.

Assistants

- Agent-like experience.
- Can call models and tools to perform tasks.
- Code-interpreter, retrieval, function calling.
- Persistent and infinitely long conversation threads. ChatGPT keeps the conversation memory for us.
- Use cases:
  - Coding assistant
  - Vacation planner
  - Voice-controlled DJ
  - Visual canvas

- See Assistants at work in the OpenAI playground.

Seed parameter: reproducible outputs

- The seed parameter ensures that the output (response) of the model to the same input is going to be the same (or very similar). This does not work 100% of times, but most of the time. It is good for testing.

Multi-modality: Vision

- Ask about the content of an image
- Text-to-image with Dall-E

Multi-modality: Audio

- Before we had just speech-to-text with Whisper.
- Now we also have text-to-speech.