

Sección 2 – Configuraciones

Para esta versión de demostración en español se han entregado dos versiones del software precargado:

Versión 1 - `conceptCurve_spanish_Biblia.zip`: Los primeros seis libros de la Biblia (BTX Biblia Textual), un documento con alrededor de 250 mil tokens divididos en 211 chunks.

Versión 2 - `conceptCurve_spanish_CCCN.zip`: La totalidad del Código Civil y Comercial de la Nación Argentina, dividido en 37 chunks con unos 220 mil tokens, el cual ChatGPT no puede leer ni sobre cual está entrenado.

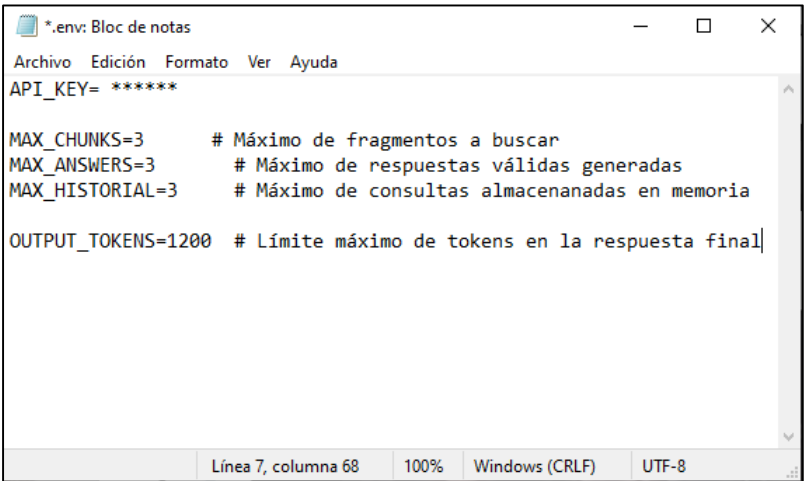
Diferencias entre versiones:

1. **En el subdirectorio `data`** donde se almacena el documento que el Software debe leer, se almacenan diferentes documentos: la versión que contiene los seis primeros libros de la Biblia y la que contiene el Código Procesal Civil y Comercial de la Nación Argentina.
2. **Configuración en el archivo `.env`** Para cada documento, se recomienda configurar el software para que examine entre el 4% y el 10% de los chunks que contiene.

Ejemplo: Dado que la Biblia está dividida en 211 chunks, una buena configuración sería `MAX_CHUNKS=11`.

Para visualizar el funcionamiento del software, puede consultar este video:
<https://www.youtube.com/watch?v=pHUmzR-8NUk>

Configuración del Software



```
*.env: Bloc de notas
Archivo Edición Formato Ver Ayuda
API_KEY= *****

MAX_CHUNKS=3      # Máximo de fragmentos a buscar
MAX_ANSWERS=3     # Máximo de respuestas válidas generadas
MAX_HISTORIAL=3   # Máximo de consultas almacenadas en memoria

OUTPUT_TOKENS=1200 # Límite máximo de tokens en la respuesta final
```

En el directorio raíz donde se ha instalado el software, se encontrará el archivo de configuraciones. Las opciones de configuración cinco:

- (1) `API_KEY` Para que el software funcione, el usuario debe contratar una API Key. Inicialmente este software está diseñado para trabajar con OpenAI, pero puede adaptarse a otros proveedores de IA, incluyendo opciones offline. Los desarrolladores pueden personalizar esta configuración modificando el archivo `smartFunctions.js`.
- (2) `MAX_CHUNKS` determina el número de chunks que el software buscará para responder a la consulta realizada por el usuario. Por experiencia, es buena práctica configurar este parámetro entre el 4% y el 10% de los chunks en los que se encuentra dividido el documento a consultar.

Ejemplo: el documento del Código Civil y Comercial de la Nación Argentina está dividida en 37 chunks, sobre lo cual es bueno configurar MAX_CHUNKS=3

- (3) **MAX_ANSWERS** determina el número máximo de respuestas positivamente encontradas antes de detener la búsqueda. Como práctica recomendable es bueno dejar este número en 2 o 3, pero depende del documento consultado.

Ejemplo 1: Si consultamos en el Código Civil acerca de “los derechos del propietario de un inmueble” seguramente MAX_ANSWERS=2 es suficiente.

Ejemplo 2: Pero si consultamos en la Biblia “enumera la cantidad de veces que los hijos de Israel se rebelaron contra Moisés”, seguramente necesitaremos un número alto de MAX_ANSWERS, porque las referencias son muchas.

- (4) **MAX_HISTORIAL** El software no guarda la totalidad del historial de la conversación en memoria porque esa sería una práctica muy ineficiente. El software guarda un historial mínimo de consultas las cuales son suficientes para mantener el contexto de la conversación, y así mismo cada subsiguiente consulta en el contexto de las anteriores.

En la práctica MAX_HISTORIAL=3 es suficiente en todos los casos.

Un truco que el software utiliza para evitar los costos computacionales inútiles consiste en la creación de una Smart-Function que realiza el pulido de la pregunta, obteniendo un prompt perfecto que sirve a la IA para mantener el flujo de la conversación en su contexto apropiado. Por este motivo es que no se necesita utilizar todo el historial, siendo suficiente MAX_HISTORIAL=3.

¿Qué son las Smart-Function? Este es un concepto desarrollado por Daniel Bistman. Más información sobre esto se encuentra en la SECCIÓN 3 – Para Desarrolladores

También se puede consultar al Agente CC ---> <https://tinyurl.com/agente-cc>

- (5) **OUTPUT_TOKENS** Es el máximo de longitud que utilizará el software para entregar la respuesta final pulida presentándola en el Frontend.