Análisis Comparativo de Estrategias de Segmentación (Chunking)

En un sistema RAG (Retrieval-Augmented Generation), la forma en que se divide un documento grande en trozos pequeños (chunks) es fundamental para garantizar que el LLM pueda recibir el contexto relevante y no pierda información crítica.

1. Estrategia Recursiva (RecursiveCharacterTextSplitter)

Recomendada por defecto

Esta es la mejor estrategia para la mayoría de los casos y es la que usamos como base en la opción "Recursive (Recomendada)".

Aspecto	Descripción
Funcionamiento	Intenta preservar la estructura lógica y el contexto. Utiliza una lista de separadores (por defecto: \n\n, \n, .) y solo recurre a cortar por un número fijo de caracteres si no encuentra un separador adecuado.
Ventajas	**Preserva el Contexto Lógico.** Reduce drásticamente la probabilidad de que una frase o idea crítica sea cortada a la mitad por un límite arbitrario de caracteres. Genera *chunks* más coherentes.
Desventajas	Es ligeramente más lenta de procesar inicialmente y, a veces, puede crear *chunks* que son demasiado grandes si no se encuentra ningún separador lógico.
Mejor para	Documentos heterogéneos, ensayos, párrafos largos, libros de texto con buena estructura narrativa.

En resumen: La estrategia recursiva prioriza la integridad semántica de la información. Esto es crucial porque los modelos de *embedding* (el paso previo a la búsqueda) y el LLM

entienden mejor el contexto si las frases y los párrafos están completos.

2. Estrategia de Tamaño Fijo (Fixed Size Splitter)

Esta estrategia es la más simple y, a menudo, la más peligrosa para la calidad del RAG.

Aspecto	Descripción
Funcionamiento	Divide el texto en trozos de tamaño N (por ejemplo, 1500 caracteres) de manera estricta, utilizando solo un carácter de superposición (*overlap*) para conectar el final de un *chunk* con el inicio del siguiente.
Ventajas	Es la más rápida y predecible. Garantiza que todos los *chunks* sean exactamente del mismo tamaño máximo.
Desventajas	**Riesgo de Contexto Roto.** Un corte arbitrario a mitad de una frase o incluso a mitad de una palabra es muy común. Si el concepto clave se divide entre dos *chunks*, es posible que el sistema no recupere ninguno de ellos, lo que lleva a respuestas erróneas o "alucinaciones".
Mejor para	Datos extremadamente estructurados, como líneas de código o archivos CSV, donde la pérdida de contexto narrativo es menos relevante que mantener una estructura fija.

3. Estrategia Estructural o Específica del Formato

Aunque es posible que la tercera estrategia que utilizaste sea una simple variación de las dos anteriores (por ejemplo, un tamaño fijo con un *overlap* mayor o menor), las mejores alternativas a la Recursiva son las que entienden el *formato*.

Aspecto	Descripción
Funcionamiento	Utiliza reglas de formato (como un *parser*

	de Markdown, PDF o LaTeX) para segmentar el texto. Por ejemplo, siempre corta cuando se encuentra un nuevo título (#, ##) o una nueva sección.
Ventajas	**Máxima Coherencia Estructural.** Mantiene la sección completa de un documento junta, lo que facilita el razonamiento jerárquico del LLM.
Desventajas	Requiere que el documento de origen tenga un formato limpio y coherente (ej: que todos los títulos de sección usen el mismo nivel de encabezado).
Mejor para	Manuales, documentación técnica, código fuente, archivos Markdown.

Conclusión y Recomendación

La mejor estrategia es la Recursiva.

Te recomiendo seguir usando la **Estrategia Recursiva (Recomendada)** como tu opción principal, ya que ofrece el mejor equilibrio entre granularidad y preservación del contexto para la mayoría de los documentos en formato PDF/texto.

Si notas que en alguna de tus pruebas el LLM necesita ver el "panorama completo" (es decir, el contexto es muy amplio), te sugiero experimentar con los parámetros de la estrategia recursiva:

1. **Aumentar el CHUNK_SIZE:** Haz los trozos más grandes (por ejemplo, de 1500 a 2000) para incluir más contexto.
2. **Aumentar el CHUNK_OVERLAP:** Aumenta la superposición (de 300 a 500) para asegurar una mejor transición entre los trozos.