5 Levels Of Text Splitting

1. Introduction

Semantic text chunking is an advanced Natural Language Processing (NLP) technique used to divide large text documents into smaller chunks based on **meaning and contextual similarity** rather than fixed size rules. This approach is especially important in modern applications involving Large Language Models (LLMs), Retrieval-Augmented Generation (RAG), and semantic search systems.

Unlike traditional splitting methods, semantic chunking ensures that each chunk preserves topic continuity and contextual relevance.

2. Objective

The objectives of this project are:

- To understand semantic-aware text splitting
- To implement chunking using embedding similarity
- To preserve contextual coherence in text chunks
- To prepare text for downstream LLM and RAG tasks

3. Dataset Description

The dataset used consists of:

- Large textual documents
- Paragraphs or sentences forming coherent topics

The text is processed sequentially to detect semantic shifts.

4. Methodology

4.1 Text Preprocessing

- Text cleaning and normalization
- Sentence or paragraph segmentation

4.2 Embedding Generation

- Each text segment is converted into a vector embedding
- Pre-trained embedding models are used for semantic representation

4.3 Similarity Measurement

- Cosine similarity is computed between consecutive embeddings
- Similarity scores indicate contextual closeness

4.4 Chunk Formation

- A new chunk is created when similarity falls below a defined threshold
- Semantically similar segments are grouped together

5. Working Principle

1. Input text is segmented into smaller units
2. Embeddings are generated for each unit
3. Similarity between adjacent segments is calculated
4. Semantic boundaries determine chunk breaks

6. Results

- Generated chunks preserve topic continuity
- Improved semantic relevance compared to rule-based methods
- Better performance in retrieval-based tasks

7. Advantages

- High semantic preservation
- Ideal for RAG and semantic search systems
- Produces contextually meaningful chunks

8. Limitations

- Computationally expensive
- Requires embedding models
- Threshold selection impacts chunk quality

9. Applications

- Retrieval-Augmented Generation (RAG)
- Document indexing
- Semantic search engines
- Intelligent chatbots

10. Conclusion

Semantic text chunking effectively improves text preprocessing by preserving meaning and contextual relevance. It is a crucial technique for modern NLP and LLM-based systems where semantic accuracy is critical.

11. Future Scope

- Adaptive similarity thresholds
- Hybrid recursive + semantic chunking
- Integration with vector databases
- Real-time chunking for streaming data