Chunking for RAG Systems

Why It Exists, How It Works, and How It Breaks Everything

Retrieval-Augmented Generation (RAG)

Why Chunking?

In RAG systems:

Good chunking = good answers
Bad chunking = hallucinations

Chunking affects:

- Retrieval accuracy
- Context relevance
- Answer quality

What Is Chunking?

Chunking is the process of breaking large documents into **small, meaningful pieces** of text

Each chunk becomes:

- One embedding
- One retrievable unit

Why Chunking Exists?

- Documents are too large
- LLM context window is limited
- Embeddings work best on small text

One big document → poor retrieval

What Happens Without Chunking?

Imagine asking:

“What is the attendance policy?”

If the entire handbook is one chunk:

- Similarity is diluted
- Retrieval is inaccurate
- Wrong context is sent to LLM

Chunking as a Search Problem

Think of chunking as:

“How can I split text so that **one chunk answers one question**?”

If a chunk can answer:

- exactly one type of question

That’s a good chunk.

Chunk Size: Too Large

Example

Chunk = 3 pages of text

Problems:

- Contains many topics
- Embedding becomes “average”
- Retrieval is vague
- LLM gets irrelevant context

Chunk Size: Too Small

Example

Chunk = 1 sentence

Problems:

- Loses context
- Definitions split from explanations
- Multiple chunks needed to answer one question

Chunk Size: Just Right

Rule of Thumb

- 300–1000 characters
- Or 150–300 words
- Keep related ideas together

Goal:
One chunk ≈ one concept

Why Chunk Overlap Exists

Problem Without Overlap

If a sentence is split:

- Half goes into chunk A
- Half into chunk B

Meaning is lost.

Chunk Overlap Explained

Overlap means:

- Repeating some text between chunks

Example:

- Chunk 1: sentences 1–10
- Chunk 2: sentences 8–18

This preserves context.

Typical Overlap Values

Chunk Size	Overlap
300 chars	50–80
600 chars	100–150
1000 chars	150–200

No overlap → fragile RAG.

Heading-Based Chunking

What It Is

- Split text by headings
- Keep sections together

Example:

Attendance Policy

(text...)

Why Heading-Based Chunking Is Powerful

- Preserves structure
- Keeps definitions + rules together
- Matches how humans read documents

Perfect for:

- Policies
- Manuals
- Handbooks

Sentence-Based Chunking

What It Is

- Split text sentence by sentence
- Group sentences until size limit

Good for:

- Plain articles
- Blogs
- Unstructured text

Heading vs Sentence-Based Chunking

Heading-Based	Sentence-Based
Structured	Flexible
Clean retrieval	More generic
Best for policies	Best for articles
Requires headings	No headings needed

What Happens When Chunking Is Bad

- Answer mentions wrong section
- LLM mixes topics
- Retrieval returns unrelated chunks
- Hallucinations increase

Real Example of Bad Chunking

Chunk contains:

- Attendance rules
- Grading system
- Exam policy

Question:
“What happens if I miss classes?”

LLM answer:
Mentions grading and exams

Chunking Is NOT One-Size-Fits-All

Chunking depends on:

- Document type
- Language
- Question style
- Domain (legal, medical, academic)

Always test and adjust.

How to Know Chunking Is Working

Ask:

- Are retrieved chunks clearly relevant?
- Can one chunk answer the question?
- Do answers cite correct sections?

If yes → chunking is good.

Key Takeaways

- Chunking is the foundation of RAG
- Chunk size balances context and focus
- Overlap preserves meaning
- Headings make chunking smarter
- Bad chunking breaks everything

What’s Next

Embeddings

- What they represent
- Why similarity works
- How chunking affects embeddings