# Chunking


- What is recursive chunking
- Chunking controls
- Choosing chunk size and overlap
  - Additional considerations:


Chunking is the process of splitting large data into smaller segments before embedding them for search. AI Search uses **recursive chunking**, which breaks your content at natural boundaries (like paragraphs or sentences), and then further splits it if the chunks are too large.

# What is recursive chunking

Recursive chunking tries to keep chunks meaningful by:

- **Splitting at natural boundaries:** like paragraphs, then sentences.
- **Checking the size:** if a chunk is too long (based on token count), it's split again into smaller parts.

This way, chunks are easy to embed and retrieve, without cutting off thoughts mid-sentence.

# Chunking controls

AI Search exposes two parameters to help you control chunking behavior:

- **Chunk size**: The number of tokens per chunk. The option range may vary depending on the model.
- **Chunk overlap**: The percentage of overlapping tokens between adjacent chunks.
  - Minimum: `0%`
  - Maximum: `30%`

These settings apply during the indexing step, before your data is embedded and stored in your search index.

# Choosing chunk size and overlap

Chunking affects both how your content is retrieved and how much context is passed into the generation model. Try out this external chunk visualizer tool to help understand how different chunk settings could look.

# Additional considerations:

- **Index size:** Smaller chunk sizes produce more chunks and more total vectors. Refer to the AI Search limits to ensure your configuration stays within instance limits.
- **Generation model context window:** Generation models have a limited context window that must fit all retrieved chunks (`max_num_results` × `chunk size`), the user query, and the model's output. Be careful with large chunks or high `max_num_results` values to avoid context overflows.
- **Ingestion cost:** Overlapping text is counted in every chunk it appears in, so a higher chunk overlap can increase billable ingestion tokens. Refer to How ingestion tokens are counted.
- **Cost and performance:** Larger chunks and higher `max_num_results` settings result in more tokens passed to the model, which can increase latency and cost. You can monitor this usage in AI Gateway.

---

□□□ #2
□ □□□ □□□ 2026-10-31 23:46:53 UTC
□ □□□ □□□ 2026-10-31 23:46:54 UTC