Recursive Character Text Splitter node documentation

# Recursive Character Text Splitter node#

The Recursive Character Text Splitter node splits document data recursively to keep all paragraphs, sentences then words together as long as possible.

On this page, you'll find the node parameters for the Recursive Character Text Splitter node, and links to more resources.

Parameter resolution in sub-nodes

Sub-nodes behave differently to other nodes when processing multiple items using an expression.

Most nodes, including root nodes, take any number of items as input, process these items, and output the results. You can use expressions to refer to input items, and the node resolves the expression for each item in turn. For example, given an input of five name values, the expression {{ $json.name }} resolves to each name in turn.

name

{{ $json.name }}

In sub-nodes, the expression always resolves to the first item. For example, given an input of five name values, the expression {{ $json.name }} always resolves to the first name.

name

{{ $json.name }}

## Node parameters#

• Chunk Size: Enter the number of characters in each chunk.

• Chunk Overlap: Enter how much overlap to have between chunks.

## Templates and examples#

by n8n Team

by Jimleuk

by David Roberts

## Related resources#

Refer to LangChain's text splitter documentation and LangChain's recursively split by character documentation for more information about the service.

View n8n's Advanced AI documentation.

## AI glossary#

• completion: Completions are the responses generated by a model like GPT.

• hallucinations: Hallucination in AI is when an LLM (large language model) mistakenly perceives patterns or objects that don't exist.

• vector database: A vector database stores mathematical representations of information. Use with embeddings and retrievers to create a database that your AI can access when answering questions.

• vector store: A vector store, or vector database, stores mathematical representations of information. Use with embeddings and retrievers to create a database that your AI can access when answering questions.