Analytics Vidhya

Mastering RAG

Character Text Chunking in RAG

Clouds come floating into my life, no longer to carry rain or usher storm, but to add color to my sunset sky.

Upload .txt

Splitter: Character Splitter

Chunk Size: 35

Chunk Overlap: 0

Total Characters: 109

Number of chunks: 4

Average chunk size: 27.3

Clouds come floating into my life, no longer to carry rain or usher storm, but to add color to my sunset sky.

Splitter: Character Splitter

Chunk Size: 1000

Chunk Overlap: 200

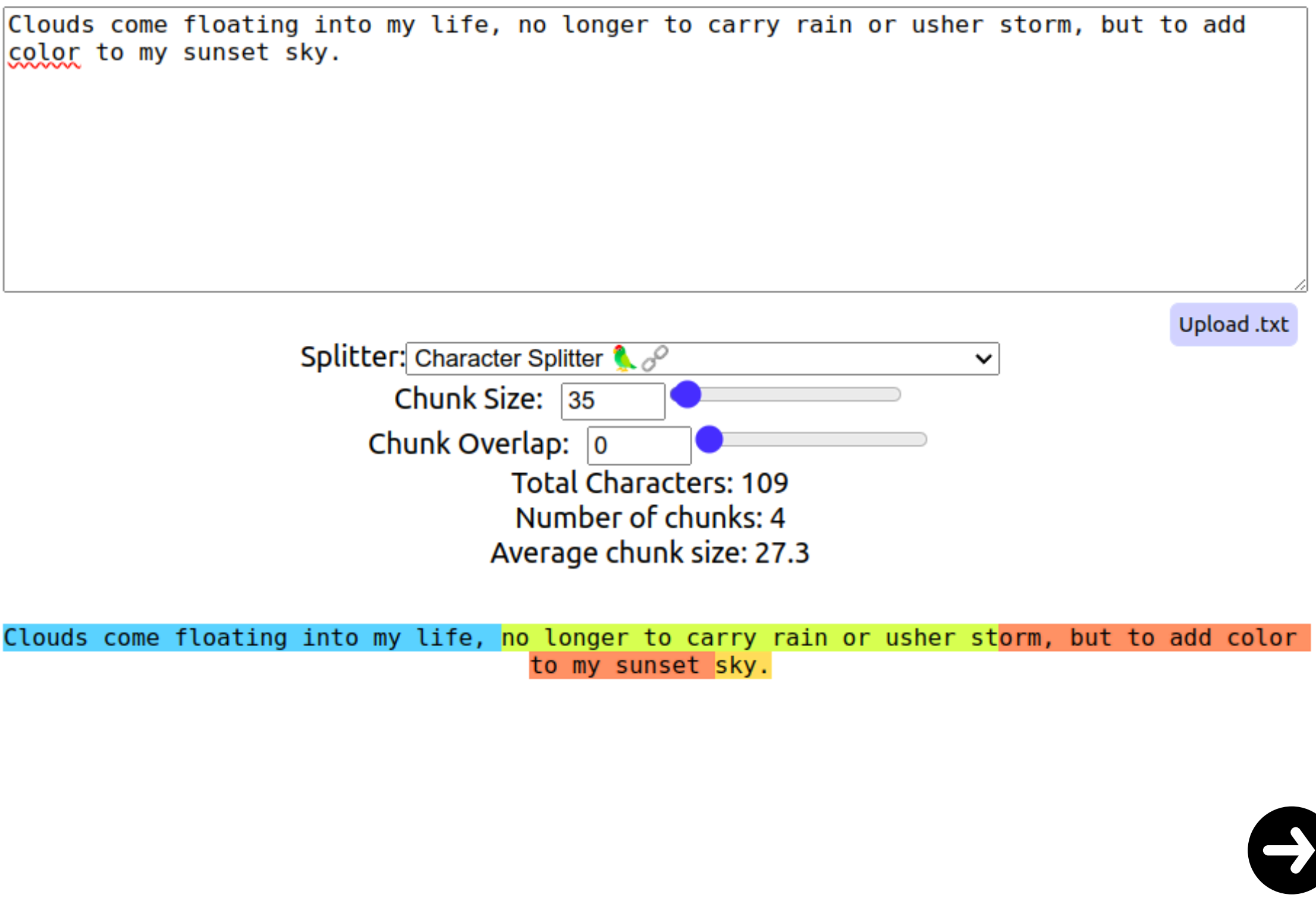
Total Characters: 47939

Number of chunks: 48

Average chunk size: 998.7

This method is one of the simplest approaches to chunking or splitting text. It divides the text into fixed-sized chunks of N characters, regardless of the content or structure. While it is a basic technique, it serves as an excellent starting point for understanding the fundamentals of text chunking and how it works in practice.

This approach is easy and simple to use; however, it is very rigid and does not take into account the structure of your text.



```
text = "Clouds come floating into my life,
no longer to carry rain or usher storm, but
to add color to my sunset sky."
chunks = []
chunk_size = 35
chunk_overlap = 5 # Characters
# Run through the text with the length of
your text and iterate every chunk_size,
# considering the overlap for the starting
position of the next chunk.
for i in range(0, len(text) - chunk_size +
1, chunk_size - chunk_overlap):
    chunk = text[i:i + chunk_size]
    chunks.append(chunk)
chunks
```

Output

['Clouds come floating into my life, ',
 'ife, no longer to carry rain or ush',
 'r usher storm, but to add color to ']

Explanation:

Input Text:

- A string variable text contains a sentence.

Chunks List Initialization:

- chunks = [] creates an empty list to store text segments.

Chunking Parameters

- chunk_size = 35: Defines the length of each chunk to be 35 characters.
- chunk_overlap = 5: Specifies that each chunk will overlap with the previous one by 5 characters.

Chunking Process:

- The for loop iterates through the text using a step size of chunk_size – chunk_overlap, meaning new chunks will start every 30 characters but will include the last 5 characters from the previous chunk.

- The loop range is determined by len(text) – chunk_size + 1, ensuring it doesn’t go beyond the text length.

- In each iteration, a substring of length chunk_size is extracted from the text and added to the chunks list.

Explanation of the Overlapping Mechanism

Step Size Calculation:

Document embeddings represent entire documents, considering overall topic structure and relationships between words and sentences.

- The loop iterates with a step of chunk_size – chunk_overlap, which means:

  35−5=30.

- This means after processing the first 35 characters, the next chunk starts 30 characters after the first one, causing a 5-character overlap.

Let’s analyze how the loop runs with the given values:

First chunk (index 0 to 35):
Extracts the substring “Clouds come floating into my life, “.

The loop then moves forward by 30 characters.

Second chunk (index 30 to 65):
Extracts the substring “ife, no longer to carry rain or ush”.

Notice how the last 5 characters of the previous chunk (“life,”) overlap in this chunk.

Third chunk (index 60 to 95):
Extracts the substring “r usher storm, but to add color to “.

Again, there’s an overlap with the last few characters from the second chunk.

Now let’s do it with Langchain

Splitter: Character Splitter

Chunk Size: 1000

Chunk Overlap: 200

Total Characters: 47939

Number of chunks: 48

Average chunk size: 998.7

Madam Speaker, Madam Vice President, our First Lady and Second Gentleman. Members of Congress and the Cabinet. Justices of the Supreme Court. My fellow Americans.

Last year COVID-19 kept us apart. This year we are finally together again.

Tonight, we meet as Democrats Republicans and Independents. But most importantly as Americans.

With a duty to one another to the American people to the Constitution.

And with an unwavering resolve that freedom will always triumph over tyranny.

Six days ago, Russia’s Vladimir Putin sought to shake the foundations of the free world thinking he could make it bend to his menacing ways. But he badly miscalculated.

He thought he could roll into Ukraine and the world would roll over. Instead he met a wall of strength he never imagined.

He met the Ukrainian people.

From President Zelenskyy to every Ukrainian, their fearlessness, their courage, their determination, inspires the world.

Groups of citizens blocking tanks with their bodies. Everyone from students to retirees teachers turned soldiers defending their homeland.

In this struggle as President Zelenskyy said in his speech to the European Parliament “Light will win over darkness.” The Ukrainian Ambassador to the United States is here tonight.

Let each of us here tonight in this Chamber send an unmistakable signal to Ukraine and to the world.

Please rise if you are able and show that, Yes, we the United States of America stand with the Ukrainian people.

Throughout our history we’ve learned this lesson when dictators do not pay a price for their aggression they cause more chaos.

```
%pip install -qU langchain-text-splitters
```

This command installs the langchain-text-splitters library, which is used for splitting long pieces of text into smaller chunks.

The -q flag suppresses installation output, and -U ensures that the latest version is installed.

```
# Load an example document
with open("state_of_the_union.txt") as f:
  state_of_the_union = f.read()
```

- Opens the file state_of_the_union.txt and reads its entire content into the variable state_of_the_union as a string.
- This document is presumably the transcript of a U.S. State of the Union address.

```
text_splitter = CharacterTextSplitter(
    separator="\n\n",
    chunk_size=1000,
    chunk_overlap=200,
    length_function=len,
    is_separator_regex=False,
)
```

This code sets up a CharacterTextSplitter object with the following parameters:

- **separator=”\n\n”**

  The document is split by double newline characters (\n\n), which typically indicate paragraph breaks in text files.

- **chunk_size=1000**

  Each text chunk will contain approximately 1000 characters.

- **chunk_overlap=200**

  There will be a 200-character overlap between consecutive chunks to ensure context continuity when processing the text.

- **length_function=len**

  Specifies that the length of each chunk is calculated using Python's built-in len() function, which measures the number of characters.

- **is_separator_regex=False**

  Indicates that the separator provided ("\n\n") is a literal string and not a regular expression.

```
texts =
text_splitter.create_documents([state_of_the
_union])
print(texts[0])
```

chunk_overlap=200 There will be a 200-character overlap between consecutive chunks to ensure context continuity when processing the text. length_function=len Specifies that the length of each chunk is calculated using Python's built-in len() function, which measures the number of characters. is_separator_regex=False Indicates that the separator provided ("\n\n") is a literal string and not a regular expression.

```
page_content='Madam Speaker, Madam Vice President, our First Lady and Second Gentleman. Members of Congress and the Cabinet. Justices of the Supreme Court. My fellow Americans.

Last year COVID-19 kept us apart. This year we are finally together again.

Tonight, we meet as Democrats Republicans and Independents. But most importantly as Americans.

With a duty to one another to the American people to the Constitution.

And with an unwavering resolve that freedom will always triumph over tyranny.

Six days ago, Russia's Vladimir Putin sought to shake the foundations of the free world thinking he could make it bend to his menacing ways. But he badly miscalculated.

He thought he could roll into Ukraine and the world would roll over. Instead he met a wall of strength he never imagined.

He met the Ukrainian people.

From President Zelenskyy to every Ukrainian, their fearlessness, their courage, their determination, inspires the world.'
```

Chunking in Action:

- The content is split into paragraphs based on the double newline (\n\n) separator.
- This ensures the logical separation of ideas while maintaining readability.

Overlap Handling:

- The chunk may contain up to 200 characters from the previous chunk to preserve context.