

Anhang A

Exkurs: Die Kunst des Zerlegens – ein tieferer praktischer Einblick ins Text-Chunking

Ich habe bei der Beschreibung des RAG-Prozesses bereits kurz den Schritt des »Aufteilens« (*Splitting/Chunking*) der Informationen als Teil der Vorbereitung erklärt. Doch hinter diesem einfachen Wort verbirgt sich eine der wichtigsten Stellschrauben für die Qualität und Präzision Ihres gesamten KI-Wissensmanagements. Die Art und Weise, *wie* Sie Ihre Informationen in die gut verarbeitbaren »Karteikarten« zerlegen, hat einen fundamentalen Einfluss darauf, wie gut Ihr System später die richtigen Antworten finden kann. Es wirkt sich direkt auf die Zufriedenheit der Anwender aus, ob für sie das System nutzbar ist, also die erwarteten und richtigen Antworten auf ihre Fragen liefert, oder eben nicht.

In diesem Exkurs sehen wir uns die wichtigsten Chunking-Strategien im Detail an. Das Verständnis dieser Methoden gibt Ihnen die Kompetenz, Ihr RAG-System optimal auf die Beschaffenheit Ihrer Firmendokumente abzustimmen, also auf Ihre Informationen zusammen mit Ihren fachlichen Anforderungen.

Chunks und Vektordatenbanken: Eine perfekte Symbiose

Um zu verstehen, warum Chunking so entscheidend ist, müssen wir kurz auf das Herzstück der RAG-Suche zurückkommen: die *Vektordatenbank*.

Eine traditionelle Stichwortsuche findet Dokumente, in denen exakt Ihre Suchwörter vorkommen. Eine Vektordatenbank geht einen bedeutenden Schritt weiter: Sie ermöglicht eine *semantische Suche*. Bei dieser Art von Suche sucht das System nicht nach einzelnen Wörtern, sondern nach der *Bedeutung*. Wenn Sie nach »Urlaubsanspruch bei Teilzeit« fragen, findet sie auch einen Absatz, in dem nur von »Freistellung für Angestellte mit reduzierter Stundenzahl« die Rede ist, weil sie versteht, dass beide Phrasen inhaltlich eng verwandt sind. Das bedeutet aber auch im Umkehrschluss, dass solch ein RAG-basierter Ansatz so gut wie gar nicht die folgende Frage beantworten kann: »Wie oft kommt in meinen Dokumenten der Begriff Teilzeit vor?«

Genau hier kommt das Chunking und dessen Funktionsweise ins Spiel. Wir speichern nicht den Vektor eines ganzen 50-seitigen Dokuments in der Datenbank. Das wäre so, als würden Sie ein ganzes Buch unter einem einzigen Schlagwort ablegen. Stattdessen speichern wir die Vektoren von Hunderten kleinen Chunks. Fragt ein Nutzer etwas, findet die Datenbank genau die drei bis vier Chunks (also Absätze oder »Karteikarten«), die der Be-

deutung seiner Frage am nächsten kommen. Diese hochrelevanten Textausschnitte werden dann dem LLM als Kontext übergeben, was zu extrem genauen Antworten führt. Häufig können Sie auch die Anzahl der Chunks festlegen, die an das LLM als Kontext (also als Ihr Suchergebnis in der Vektordatenbank) weitergegeben werden sollen.

Die Wahl der richtigen Chunking-Strategie ist daher die Kunst, Ihre Dokumente so zu zerlegen, dass jeder Chunk eine in sich geschlossene, sinnvolle Informationseinheit darstellt. Die Strategien des Chunkings werden sich je nach Dokument unterscheiden. Wenn Sie den *EU AI Act* mit dem *BOSCH*-Werkzeugkatalog vergleichen, werden Sie sehr schnell auf unterschiedliche Strategien für das Chunking kommen, da der Aufbau der Dokumente komplett unterschiedlich ist.

Für die Herausarbeitung einer Chunking-Strategie und die Visualisierung, wie diese Ihren Text zerlegt, ist eine grafische Aufbereitung sehr gut geeignet. In Abbildung A.1 sehen Sie einen Screenshot des Projekts *Chunk Viz*, das ich sehr empfehlen kann. Es hilft Ihnen dabei, verschiedene Strategien der Textaufteilung zu verproben. *Chunk Viz* ist auf GitHub verfügbar, sodass Sie es bei sich lokal einrichten können und Sie Ihre Texte nicht in die Online-Demo-Version einfügen müssen: <https://github.com/gkamradt/ChunkViz>

In Abbildung A.1 sehen Sie ein Beispiel für die Arbeit mit *Chunk Viz*.

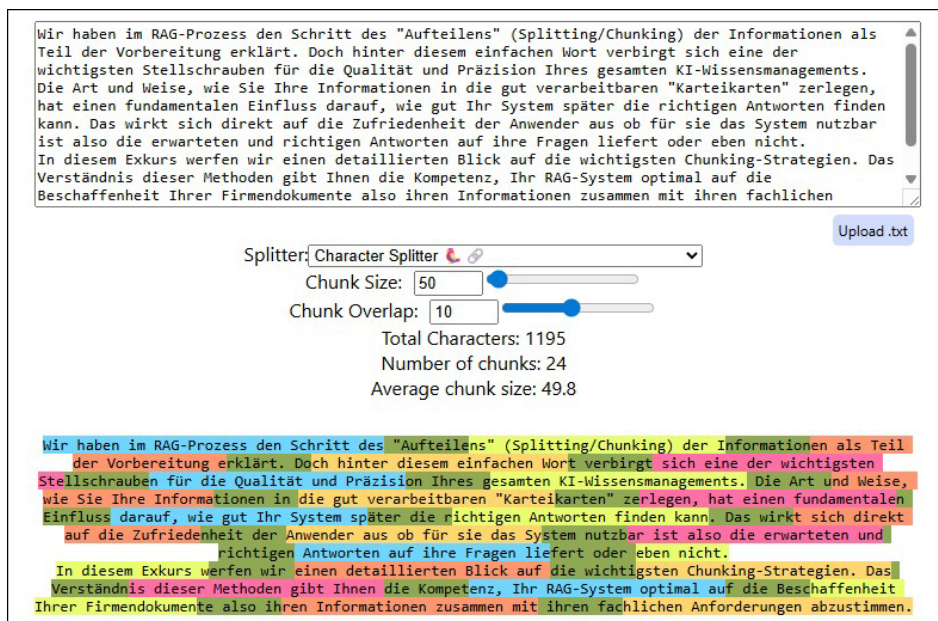


Abbildung A.1: *Chunk-Viz*-Beispieltext

Die wichtigsten Parameter: Größe und Überlappung

Bevor wir die Strategien betrachten, müssen wir zwei Grundbegriffe klären, die Sie auch in den RAG-Einstellungen von Enterprise-Ready-Lösungen wiederfinden:

- **Chunk-Größe (Chunk Size):** Legt die maximale Länge eines Chunks fest, meist in Zeichen oder Tokens. (Ein Token ist die kleinste Texteinheit eines LLM-Modells.)
- **Chunk-Überlappung (Chunk Overlap):** Definiert, wie viele Zeichen vom Ende eines Chunks am Anfang des nächsten Chunks wiederholt werden. Dies ist ein wichtiger Trick, um zu verhindern, dass ein Satz oder Gedanke genau an der Schnittstelle zweier Chunks auseinandergerissen wird und auf diese Weise der Kontext, also die Bedeutung, verloren geht.

Die 5 wichtigsten Chunking-Strategien in der Praxis

Schauen wir uns nun die gängigsten Methoden an, um aus einem langen Text sinnvolle Chunks zu erzeugen.

1. Chunking mit fester Größe (Fixed-Size)

Dies ist die einfachste und schnellste Methode. Sie funktioniert wie ein digitaler Papierschneider, der einen Text ohne Rücksicht auf Inhalt oder Struktur in exakt gleich große Teile zerlegt.

- **Analogie:** Der Papierschneider. Er ist schnell und simpel, schneidet aber rücksichtslos mitten durch Wörter und Sätze, wenn diese ihm in den Weg kommen. Beispiel (Größe: 120 Zeichen, Überlappung: 30 Zeichen, *Überlappung in kursiv*):
- **Originaltext:** »Die Gründung der Europäischen Union war eine direkte Antwort auf die verheerenden Kriege der ersten Hälfte des 20. Jahrhunderts. Die Vision der Gründerväter wie Robert Schuman und Konrad Adenauer war es, durch wirtschaftliche Verflechtung einen dauerhaften Frieden zu sichern. Den entscheidenden Anstoß gab die Schuman-Erklärung von 1950, die zur Gründung der Europäischen Gemeinschaft für Kohle und Stahl (EGKS) führte. Indem man die kriegswichtigen Industrien Kohle und Stahl einer gemeinsamen Kontrolle unterstellte, wurde ein weiterer Konflikt zwischen Frankreich und Deutschland materiell unmöglich gemacht. Dieser erste erfolgreiche Schritt ebnete den Weg für die Römischen Verträge von 1957, die mit der Schaffung der Europäischen Wirtschaftsgemeinschaft (EWG) die Zusammenarbeit auf viele weitere Wirtschaftsbereiche ausweiteten und das Fundament für die heutige Union legten.«
- **Erzeugte Chunks:** *Chunk 1:* Die Gründung der Europäischen Union war eine direkte Antwort auf die verheerenden Kriege der ersten Hälfte des 20. Jahrhundert; *Chunk 2:* e der ersten Hälfte des 20. Jahrhunderts. Die Vision der Gründerväter wie Robert Schuman und Konrad Adenauer war es, durch; *Chunk 3:* Schuman und Konrad Adenauer war es, durch wirtschaftliche Verflechtung einen dauerhaften Frieden zu sichern. Den entscheid ... und so weiter.

Fazit: Das ist einfach und schnell, aber rücksichtslos gegenüber der Bedeutung im Text. Durch das sture Zerschneiden von Sätzen und sogar von Wörtern geht oft der semantische Kontext verloren. Dieses Vorgehen ist nur für sehr unstrukturierte Texte ohne klare Gliederung geeignet.

2. Rekursives Chunking (Recursive Character Text Splitter)

Diese Methode ist deutlich intelligenter. Sie versucht, den Text entlang einer vordefinierten Hierarchie von Trennzeichen zu spalten. Sie sucht zuerst nach dem wichtigsten Trennzeichen (z. B. doppelte Zeilenumbrüche für Absätze). Wenn die so entstehenden Chunks immer noch zu groß sind, wendet sie das nächste Trennzeichen an (z. B. Punkte für Sätze), bis die gewünschte Chunk-Größe erreicht ist.

- **Analogie:** Der intelligente Editor. Er versucht zuerst, an den Satzenden zu schneiden. Nur wenn ein Satz zu lang ist, schneidet er am Komma. Er respektiert die natürliche Struktur des Textes so lange wie möglich.
- **Beispiel (Größe: 150, Überlappung: 40)**
- **Erzeugte Chunks:** *Chunk 1:* Die Gründung der Europäischen Union war eine direkte Antwort auf die verheerenden Kriege der ersten Hälfte des 20. Jahrhunderts; *Chunk 2: Hälfte des 20. Jahrhunderts.* Die Vision der Gründerväter wie Robert Schuman und Konrad Adenauer war es, durch wirtschaftliche Verflechtung einen dauerhaften Frieden zu sichern; *Chunk 3: Frieden zu sichern.* Den entscheidenden Anstoß gab die Schuman-Erklärung von 1950, die zur Gründung der Europäischen Gemeinschaft für Kohle und Stahl (EGKS) führte. ... und so weiter.

Fazit: Das rekursive Chunking ist ein exzellenter Allrounder und oft die Standardeinstellung in RAG-Systemen. Es bewahrt den semantischen Kontext viel besser als die Fixed-Size-Methode und ist insgesamt ein sehr guter Kompromiss aus Qualität und Geschwindigkeit.

3. Dokumentenbasiertes Chunking

Diese Strategie nutzt die explizite Struktur eines Dokuments, um Chunks zu erstellen. Bei einem Markdown-Dokument (.md) würde Sie beispielsweise den Text anhand von Überschriften (#, ##), Listen oder Code-Blöcken aufteilen.

Analogie: Der Inhaltsverzeichnis-Nutzer. Er ignoriert die Länge der Abschnitte und orientiert sich ausschließlich an der vom Autor vorgegebenen Gliederung, um logische Einheiten zu schaffen.

Beispiel (Strukturierter Text), Originaltext:

- **Die Vision des Friedens:** Die Gründung der Europäischen Union war eine direkte Antwort auf die verheerenden Kriege der ersten Hälfte des 20. Jahrhunderts ...
- **Die erste Säule: EGKS** – Den entscheidenden Anstoß gab die Schuman-Erklärung von 1950, die zur Gründung der Europäischen Gemeinschaft für Kohle und Stahl (EGKS) führte ...
- **Die zweite Säule: EWG** – Dieser erste erfolgreiche Schritt ebnete den Weg für die Römischen Verträge von 1957 ...

Beispiel (Strukturierter Text), Erzeugte Chunks:

- *Chunk 1:* # Die Vision des Friedens\nDie Gründung der Europäischen Union war eine direkte Antwort ...;

- *Chunk 2: ## Die erste Säule: EGKS*\nDen entscheidenden Anstoß gab die Schuman-Erklärung ...;
- *Chunk 3: ## Die zweite Säule: EWG*\nDieser erste erfolgreiche Schritt ebnete den Weg ...

Fazit: Das dokumentenbasierte Chunking ist die beste Methode für gut strukturierte Dokumente wie Handbücher, Berichte oder gut formatierte Texte. Sie erzeugt Chunks mit sehr hoher semantischer Dichte, da die vom Autor beabsichtigte logische Trennung erhalten bleibt.

4. Semantisches Chunking

Dies ist die fortschrittlichste, aber auch rechenintensivste Methode. Hier werden keine strukturellen Trennzeichen genutzt. Stattdessen analysiert ein LLM-Modell die semantische Bedeutung der Sätze. Es gruppiert Sätze, die thematisch sehr eng zusammengehören, in einem Chunk. Wenn ein thematischer Sprung im Text erkannt wird, beginnt ein neuer Chunk.

Analogie: Der Fachexperte. Er liest den Text und fasst Abschnitte nicht nach Länge oder Formatierung zusammen, sondern nach inhaltlicher Zugehörigkeit. Er erkennt, wo ein Gedankengang endet und ein neuer beginnt.

Erzeugte Chunks (vereinfacht):

- *Chunk 1:* (enthält alle Sätze zur Gründungsvision und den Gründervätern) Die Gründung der Europäischen Union war eine direkte Antwort auf die verheerenden Kriege der ersten Hälfte des 20. Jahrhunderts. Die Vision der Gründerväter wie Robert Schuman und Konrad Adenauer war es, durch wirtschaftliche Verflechtung einen dauerhaften Frieden zu sichern;
- *Chunk 2:* (enthält alle Sätze zur EGKS als konkretem Instrument) Den entscheidenden Anstoß gab die Schuman-Erklärung von 1950, die zur Gründung der Europäischen Gemeinschaft für Kohle und Stahl (EGKS) führte. Indem man die kriegswichtigen Industrien Kohle und Stahl einer gemeinsamen Kontrolle unterstellte, wurde ein weiterer Konflikt zwischen Frankreich und Deutschland materiell unmöglich gemacht;
- *Chunk 3:* (enthält alle Sätze zur Erweiterung durch die EWG) Dieser erste erfolgreiche Schritt ebnete den Weg für die Römischen Verträge von 1957, die mit der Schaffung der Europäischen Wirtschaftsgemeinschaft (EWG) die Zusammenarbeit auf viele weitere Wirtschaftsbereiche ausweiteten und das Fundament für die heutige Union legten.

Fazit: Das semantische Chunking liefert potenziell die qualitativ hochwertigsten Chunks, da sie rein auf der Bedeutung basieren. Der Prozess ist jedoch komplex und rechenintensiv, da er fortschrittliche NLP-Modelle erfordert. Das semantische Chunking ist somit ideal für unstrukturierte Fließtexte, bei denen höchste kontextuelle Präzision gefragt ist.

Hinweis: Token-basiertes Chunking ist eine technische Variante des Fixed-Size- oder Recursive-Chunking, bei der nicht Zeichen, sondern Tokens als Maßeinheit verwendet werden. Es ist in der Praxis sehr verbreitet, aber das grundlegende Prinzip bleibt dasselbe.

Die Wahl der richtigen Strategie ist keine akademische Übung, sondern eine entscheidende Konfiguration, die Ihre Nutzerinnen bewusst treffen müssen, um die Qualität ihrer Ergebnisse bei der Arbeit mit LLMs in Kombination mit Texten zu maximieren. Für den Anfang ist die rekursive Strategie fast immer ein exzellenter Startpunkt, um sich an das Thema Chunking heranzuarbeiten.

Hugging Face stellt eine Übersicht, das sogenannte *Embedding Leaderboard*, bereit, auf der die gängigen Embedding-Modelle miteinander verglichen werden. Diese Übersicht ist ideal, um sich zu informieren und um anschließend eine ideale Auswahl des Embedding-Modells treffen zu können: <https://huggingface.co/spaces/mteb/leaderboard>