## Methodology

After the data preparation (prepared\_data.py), the system builds the indices (index.sh). It begins by activating the virtual environment, ensures the Cassandra driver is bundled, and runs app.py to configure the database schema. This script defines four Cassandra tables: one for the inverted index to store word-document relationships with frequencies, one for storing document titles and lengths, a table for overall statistics like total document count, and another for vocabulary data including TF-IDF values used in scoring.

The main indexing logic is split between the mapper and reducer. The mapper reads through each document, breaks down the text into individual words, filters out common stopwords, and emits relevant information such as terms, document IDs, and frequencies, along with metadata like the title and document length. The reducer then gathers this data, calculates global statistics like IDF, and writes all the processed information to Cassandra in batches for efficiency.

## **Demonstration**

For running need to start with: docker compose up -d

Indexing the documents

