Technical Report: Chunking and Search Pipeline

1. Overview

This report details the implementation of a chunking and search pipeline designed to process drone product data. The pipeline leverages advanced text processing techniques, including chunking, semantic search, and retrieval strategies, to enable efficient and accurate information retrieval.

2. Components of the Pipeline

2.1 Libraries and Dependencies

The pipeline utilizes the following libraries:

- chonkie: For sentence, token, and semantic chunking.
- nltk: For natural language processing tasks like tokenization and stopword removal.
- SentenceTransformer: For generating dense embeddings for semantic search.
- CrossEncoder: For reranking search results based on relevance.
- BM25Okapi: For traditional BM25-based text retrieval.
- matplotlib: For visualizing chunk distributions and semantic similarities.
- pandas: For handling and processing the product dataset.
- numpy: For numerical computations.
- dotenv: For environment variable management.

2.2 Data Loading and Preprocessing

- Input Data: The pipeline processes a CSV file (products.csv) containing product data with columns like title and category.
- Text Combination: The title and category columns are combined to create a richer context for each product.
- Cleaning: Texts are stripped of unnecessary whitespace and newline characters.

2.3 Chunking

The ChunkingPipeline class provides three chunking strategies:

Sentence Chunking:

- Splits text into sentences using SentenceChunker.
- Combines short sentences into chunks of up to 30 words.

Token Chunking:

- Splits text into overlapping chunks of 20 tokens with a 5-token overlap using TokenChunker.

Semantic Chunking:

- Splits text into semantically meaningful chunks using SemanticChunker with the all-MiniLM-L6-v2 model.
- Combines very short chunks with the previous chunk to ensure coherence.

Post-Processing:

- Removes duplicate chunks.
- Filters out empty or invalid chunks.

2.4 Retrieval Indexing

BM25 Index:

- Texts are preprocessed to remove stopwords and domain-specific stopwords.
- Tokenized chunks are indexed using BM25Okapi.

Semantic Index:

- Dense embeddings are generated for each chunk using SentenceTransformer.

2.5 Query Expansion

- Queries are expanded with synonyms based on predefined categories (e.g., "mini-toy-drone" includes synonyms like "mini", "toy", "basic").
- This ensures broader coverage during retrieval.

2.6 Retrieval Strategies

The pipeline supports four retrieval methods:

BM25 Search:

- Scores chunks based on BM25 relevance.
- Adjusts scores using category-based matching.

Semantic Search:

- Computes cosine similarity between query embeddings and chunk embeddings.
- Adjusts scores using category-based matching.

Hybrid Search:

- Combines BM25 and semantic scores using a weighted average (alpha).
- Dynamically adjusts alpha based on query type (e.g., camera-related queries favor semantic search).

Reranked Results:

- Uses CrossEncoder to rerank top results from the hybrid search.
- Applies additional normalization and feature-specific boosting.

2.7 Evaluation

Test Queries:

Example queries include "4k camera drone", "professional drone with GPS", and "mini toy drone for beginners".

Result Saving:

Results are saved to structured text files for each query and retrieval method.

Analysis:

Chunk distributions and semantic similarities are visualized using histograms and boxplots.

3. Flow of the Pipeline

Data Loading:

- The main function loads the product data from products.csv.
- Combines title and category columns to create a list of texts.

Chunking:

- The process_texts method applies sentence, token, and semantic chunking to the texts.
- Results are logged, and chunk statistics (e.g., average length, overlap) are analyzed.

Indexing:

- The build_retrieval_index method creates BM25 and semantic indices for the chunks.

Search and Evaluation:

- BM25, semantic, hybrid, and reranked searches are performed.
- Results are saved to text files.

- The evaluate_retrieval method compares the performance of different retrieval strategies.

Visualization:

- The analyze_chunks method generates visualizations for chunk distributions and 5. Strengths and Limitations

4. Conclusion

The chunking and search pipeline is a comprehensive solution for processing and retrieving drone product data. By leveraging advanced NLP techniques and domain-specific enhancements, it ensures accurate and relevant search results. Future improvements could include optimizing performance and extending the pipeline to support additional product categories.