Virtual Thread Text Simplifier

GitHub: Eric Murray G0042390

Description

The Virtual Thread Text Simplifier is a Java-based program designed to simplify input text files by replacing words with their closest semantic match based on embeddings. It leverages **structured concurrency** and **virtual threads** to enhance performance, scalability, and efficiency.

Features

- **User-Friendly Menu**: Intuitive CLI interface for specifying file paths and controlling execution.
- **Structured Concurrency**: Uses `StructuredTaskScope` to manage dependencies and ensure efficient execution of concurrent tasks.
- **Virtual Threads**: Lightweight, scalable threading model to load embeddings and Google words concurrently.
- **Text Simplification**: Processes input files to replace words not in the Google map with their closest match using cosine similarity.
- **Error Handling**: Robust validation of file paths and proper exception handling.
- **Cosine Similarity**: Calculates vector-based semantic similarity for accurate word replacements.
- **Progress Indicator**: Real-time feedback during execution to monitor progress.

Functionality

1. **File Loading**:
    a. Concurrently loads the embeddings file and the Google-1000 file using structured concurrency.
2. **Input File Simplification**:
    a. Processes input text line-by-line, splitting words and punctuation using regex.
    b. Replaces words with their closest match based on embeddings and Google words.
3. **Output File**:
    a. Writes the simplified text to an output file, preserving the original text structure and punctuation.

How It Works

- The program first loads the embeddings and Google map using `StructuredTaskScope`, ensuring proper task dependencies.
- Virtual threads are used to process each word in the input text concurrently, allowing scalable and efficient computation.
- For each word:
    - If found in the Google map, it remains unchanged.
    - If found in the embeddings map, it is replaced with the closest word from the Google map.
    - Otherwise, it is left unchanged.

References

- GeeksForGeeks - Virtual Thread and Concurrency in Java.
- Oracle Java Documentation - Structured Concurrency and API reference.
- ChatGPT - Debugging and performance optimization insights.
- Baeldung - Advanced threading concepts.