Report: Web Scraper

Author:

Murtaza Anwaar

AI Intern

MakTek

Contents

Overview

The project involves developing a web scraper designed to extract and process internal links and page content from a specified website, ensuring efficiency and preventing infinite loops.

Key Components

1. Configuration (config.py)

Defines constants such as user-agent headers, unwanted phrases, boilerplate terms, and maximum segment length for text processing.

2. Utility Functions (utils.py)

Contains functions to:

a. Remove HTML tags
b. Clean and filter text
c. Normalize and segment content

3. Scraping Logic (scraper.py)

Implements core functionality to:

a. Check for internal links to avoid infinite loops
b. Fetch and clean page content
c. Collect and process links and content
d. Save the cleaned data in JSONL format

4. Execution Script (scraper.py)

Runs the scraper with a specified base URL, manages the process of visiting pages, and updates the list of URLs to visit while ensuring only internal links are processed.

5. Dependencies (requirements.txt)

Lists required Python packages:

a. requests
b. beautifulsoup4

Improvements Over Original Scraper

Infinite Loop Prevention: The original scraper encountered issues with infinite loops by scraping all links on the website. This updated version addresses this problem by focusing only on internal links, thereby preventing unnecessary looping and improving efficiency.

Usage

1. Setup

Install dependencies with pip install -r requirements.txt.

2. Run the Scraper

Execute the scraper with python scraper.py.

Summary

The web scraper effectively extracts and processes internal links and page content, with improved efficiency and robustness by avoiding infinite loops. The project is well-organized to ensure maintainability and ease of use.