Product Discoverer - Python + FastAPI

This application is a web crawler built using Python, FastAPI, Playwright, BeautifulSoup, and aiohttp. It crawls specified domains, handles JavaScript-rendered content, and performs infinite scrolling where needed to extract product URLs.

Features

1. **Headless Browser Crawling:**
    - Uses Playwright to handle JavaScript and infinite scrolling.
    - Extracts dynamically loaded content.
2. **BFS-Based Link Extraction:**
    - Discovers unique URLs from a domain using a breadth-first search approach.
    - Filters out unwanted URLs and content types (e.g., images, PDFs).
3. **Concurrency:**
    - Supports concurrent crawling with a configurable concurrency limit.
4. **Domain Validation:**
    - Validates input domains to ensure they are reachable.
5. **Output:**
    - Stores discovered URLs in text files.
    - Allows downloading results through an API endpoint.
6. **Background Processing:**
    - Processes domains asynchronously using FastAPI's background tasks.

Setup

Prerequisites

- Python 3.8+
- pip
- Playwright and browser dependencies

Installation

Clone the repository:

```
git clone <repository_url>
```

1. cd <repository_directory>
2. Install dependencies:
   ```
   pip install -r requirements.txt
   ```
3. Install Playwright browsers:
   ```
   playwright install
   ```

Usage

Run the Application

Start the FastAPI server:

```
python main.py
```

The application will run on `localhost:8000`.

API Endpoints

1. Start Crawling

- **Endpoint:** `POST /crawl/`
- **Description:** Begins crawling the provided domains.

Payload:
```
{
  "domains": ["https://example.com", "https://anotherdomain.com"]
}
```

Response:
```
{
  "message": "Crawling started for provided domains."
}
```

2. Download Results

- **Endpoint:** `GET /download/{domain}`
- **Description:** Downloads the results file for a given domain.
- **Example:**
  ```
  curl -O http://localhost:8000/download/example.com
  ```

Configuration

- **Concurrency:** Adjust `max_concurrency` in `extract_product_urls_headless`.
- **Infinite Scroll:** Modify `max_scrolls` and `scroll_wait` in `fetch_html_headless`.

Code Overview

Core Functions

1. **`fetch_html_headless`:**
    - Uses Playwright to fetch and render a webpage.
    - Performs optional infinite scrolling to load additional content.
2. **`extract_product_urls_headless`:**
    - Implements BFS to extract links from a domain using `fetch_html_headless`.
    - Filters out unwanted paths and file types.
3. **`crawl_domain_headless`:**
    - Orchestrates the crawling process for a single domain.
4. **`validate_domains`:**
    - Ensures provided domains are valid and reachable.

API Handlers

- **`start_crawling`:** Handles domain crawling requests.
- **`download_results`:** Serves output files for download.

Output

Results are stored in text files under `~/Desktop/product-discoverer/output_files`. Each file corresponds to a domain and contains the list of discovered URLs.

Dependencies

- **FastAPI:** Framework for building APIs.
- **Playwright:** For headless browser operations.
- **BeautifulSoup:** For HTML parsing.
- **aiohttp:** For asynchronous HTTP requests.
- **Pydantic:** For request validation.

Troubleshooting

1. **Playwright Not Installed:**

Ensure Playwright and its browsers are installed:
pip install playwright

   - playwright install
2. **Permission Issues:**
   - Check write permissions for the output directory (`~/Desktop/product-discoverer/output_files`).
3. **Dependencies Missing:**
   - Install missing dependencies using:
     pip install -r requirements.txt

Future Improvements

- Add support for advanced filtering based on user-defined rules.
- Enhance error handling and retry mechanisms.
- Introduce a dashboard for monitoring crawl progress.