News Scraper Assignment

Anand Kumar

May 2025

1 Introduction

This project implements a Python script to scrape news articles from RSS feeds of major news agencies across 25 countries, as part of an AI laboratory assignment. The script collects news data, stores it in both CSV and SQLite formats, and includes an advanced feature: language detection. The output files include the scraped articles (`news_data.csv`), a summary of articles by country and source (`news_summary.csv`), and an SQLite database (`news_data.db`). This document provides an overview of the project, setup instructions, and details on the implementation and challenges encountered.

2 Features

The script provides the following functionality:

- Scrapes news from 25 RSS feeds covering countries such as the United Kingdom, United States, Japan, Qatar, and more.
- Extracts fields: News Title, Publication Date, Source, Country, Summary/Description, News URL, and Language.
- Stores data in `news_data.csv` (CSV format) and `news_data.db` (SQLite database).
- Implements language detection using the `langdetect` library to identify the language of each article.
- Handles errors (e.g., network issues, missing fields) and removes duplicate articles based on title and URL.
- Generates a summary of collected articles by country and source, saved as `news_summary.csv`.
- Logs all operations to `news_scraper.log` for debugging and verification.

3 Advanced Feature: Language Detection

The script includes an advanced feature that detects the language of each news article based on its title and description, using the `langdetect` Python library. The detected language is added as a column (`language`) in the output files. This feature enhances data analysis by allowing users to filter or analyze articles by language. Note that language detection may fail for very short texts, in which case the language is marked as `unknown`.

4 Setup and Execution

To run the script, follow these steps:

4.1 Prerequisites

- Python 3.8 or higher.
- Required Python libraries:

  ```
  pip install feedparser pandas sqlite3 langdetect
  ```

4.2 Steps to Run

1. Clone the repository from GitHub:

   ```
   git clone https://github.com/Anand1khatri/Web-Scraper
   cd Web-Scraper
   ```

2. Install dependencies:

   ```
   pip install -r requirements.txt
   ```

3. Run the script:

   ```
   python scrab.py
   ```

4. Check the output files in the repository directory:
   - `news_data.csv`: Contains all scraped news articles.
   - `news_summary.csv`: Summary of articles by country and source.
   - `news_data.db`: SQLite database with news data.
   - `news_scraper.log`: Log file with execution details.

5 Notes

- The script includes a 0.5-second delay between RSS feed requests to respect server rate limits.
- Historical data availability depends on the RSS feed, typically covering recent articles (up to a year in some cases).
- Non-ASCII characters are handled with UTF-8 encoding to support diverse languages.

- Some feeds may fail due to network issues or invalid URLs; errors are logged in news_scraper.log.

6 Issues Encountered

During development and execution, the following challenges were observed:

- Some RSS feeds (e.g., Clarin, The Standard) failed to return data, likely due to network issues or invalid URLs, as logged in news_scraper.log. Only 8 out of 25 feeds returned data successfully in the sample run.
- Large feeds (e.g., TASS with 100 articles) increased execution time. This was mitigated by capping articles at 50 per feed and adding a 10-second timeout for requests.
- Language detection occasionally failed for short texts, resulting in unknown as the language.

7 Submission

This repository contains the following files for the assignment:

- scrab.py: The Python script for scraping news.
- news_data.csv: Sample output with 278 unique articles.
- news_summary.csv: Summary of articles by country and source.
- news_data.db: SQLite database with news data.
- README.pdf: This Pdf document (compiled as README.pdf).

The repository is hosted at https://github.com/Anand1khatri/Web-Scrapert.