CS 839 - Data Science Project - Stage 2

Team Members

- Aravind Soundararajan (soundararaj2@wisc.edu)
- Krishnan Rajagopalan (krajagopalan@wisc.edu)
- Palaniappan Nagarajan (pnagarajan3@wisc.edu)

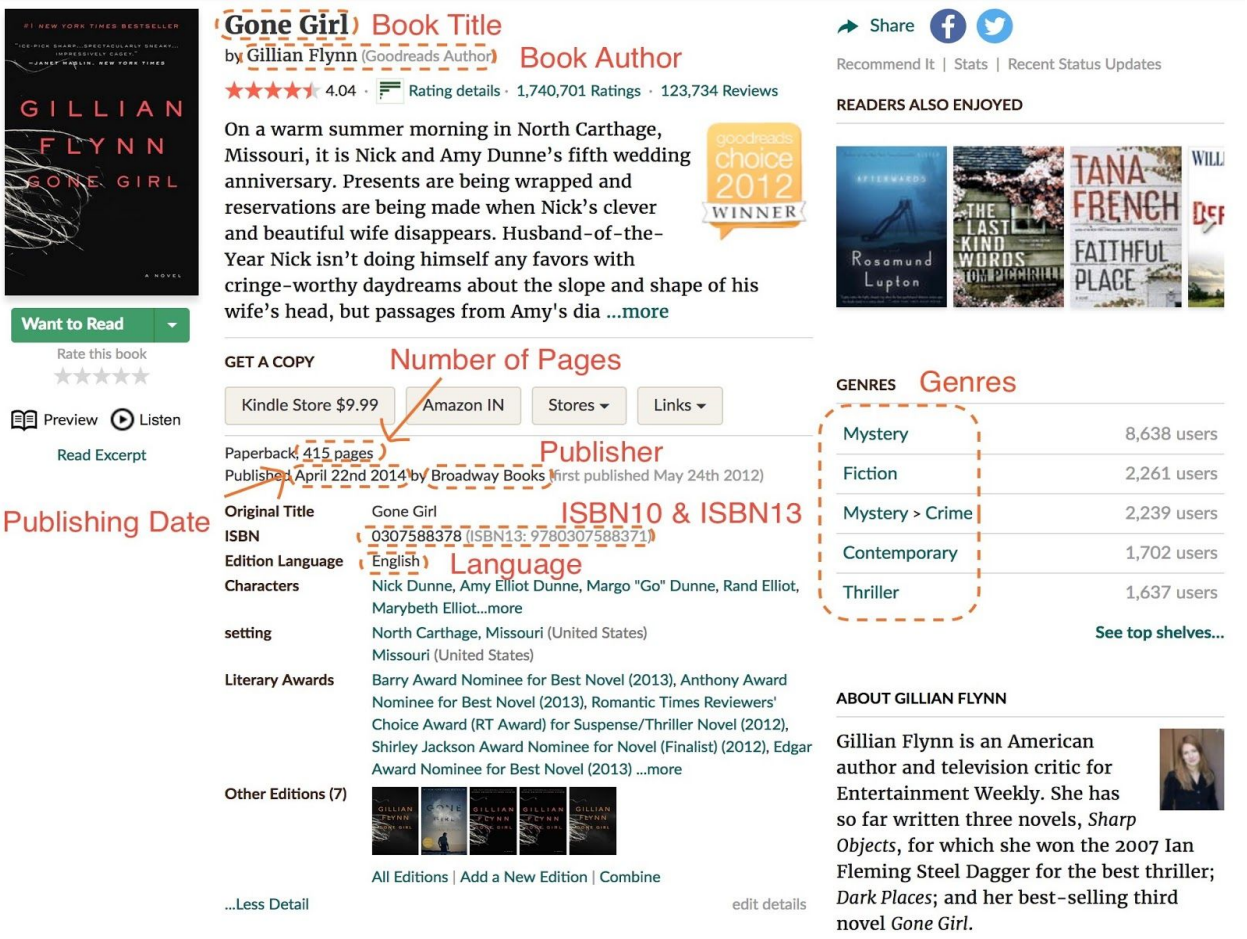
Web Sources

- Goodreads - Goodreads is a "social cataloging" website that allows individuals to freely search its database of books, annotations, and reviews. Users can sign up and register books to generate library catalogs and reading lists. They can also create their own groups of book suggestions, surveys, polls, blogs, and discussions.
- BookDepository - Book Depository is a UK-based online book seller with a large catalogue of books.

Extraction of Structured Data

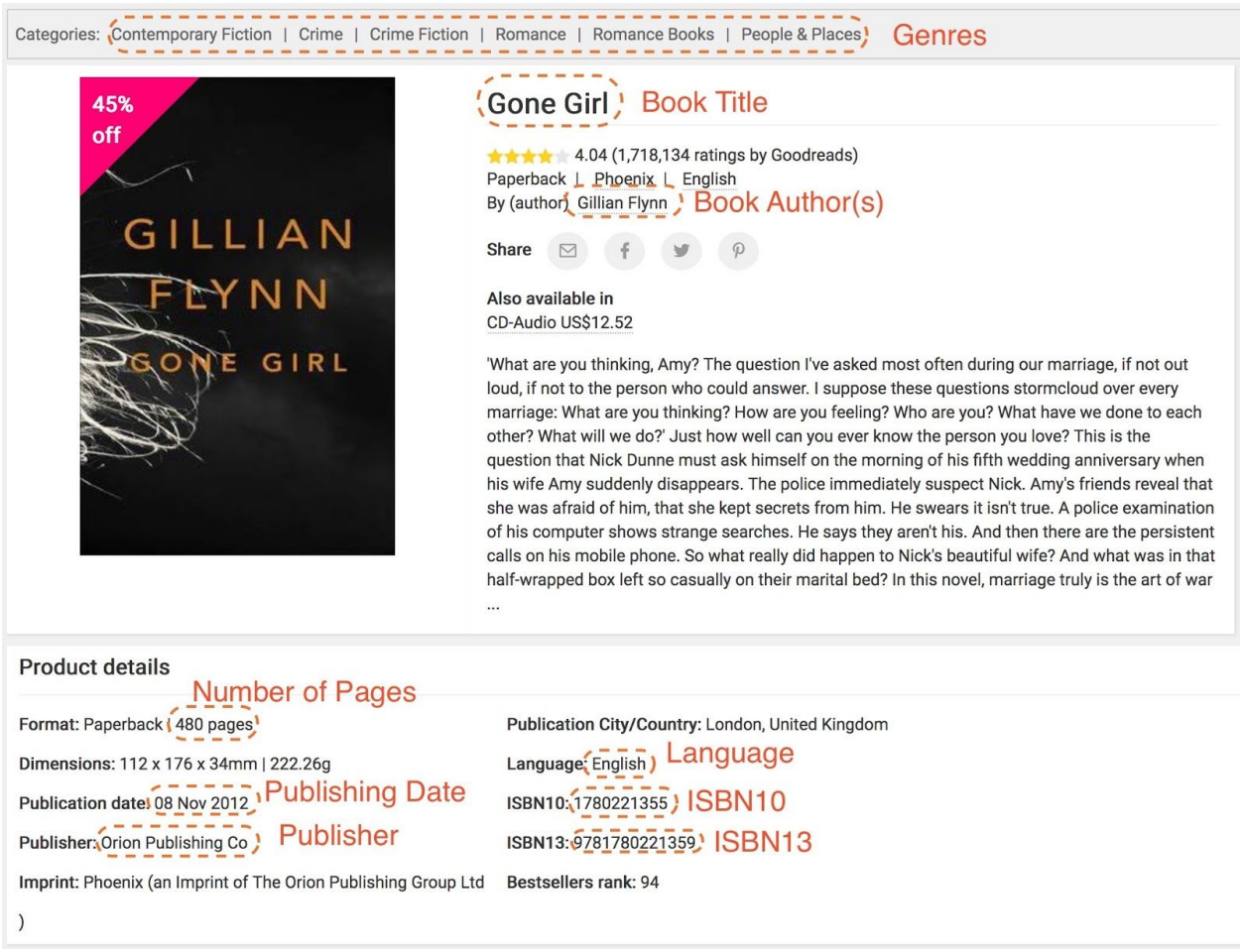
Scrapy, the open source web scraping tool was used for developing the wrapper-based extractor for both the sources.

1) **Goodreads**



The start_urls were generated to be all pages under “All Time Favorite Romance Novels” tag (genre). And xpath was used to filter out the individual links of books in the listing page. The CSS identifier of individual book links were provided to Scrapy to crawl through the various books in the results. Each of the book links were visited to crawl the details page of the corresponding book. All the information available about the book such as Author, Genres (that people have categorised it into), Publishers, Publishing date, number of pages were pulled out by using xpath identifiers for each of the attribute in the page. We have used the “Item Pipeline” by defining two pipelines. One component (BookWormsPipeline) in the pipeline uses the process_item method which performs basic cleansing of HTML data and the other component (FileWritePipeline) writes the cleaned data to CSV.

2) **BookDepository**



The start_urls were generated to be all pages under “Romance”, “Historical Romance”, “Adult-Contemporary-Romance”, “Erotic-Fiction” genres. Individual book links were identified were collected using xpath. HTTP requests to these links were triggered to collect details from within the individual book page using xpath. Also the components of the item pipeline are

common to both the spiders. The approach is quite similar to the extraction of book details from Goodreads.

Entity Type - Books
We have extracted entities of type “Book” from the two sources. Specifically, we have extracted information about books broadly in the “Romance” category. Since both the sources contained similar information about the entity (books), a common schema was decided upon before the extraction and we extracted the corresponding attributes from the two web sources.

Schema of the tables

Attribute Name	Datatype	Description
Title	String	Title of the book (including the Edition number) as given in the source from which the book details are taken.
Authors	String	Names of all the persons listed under the author’s section along with the authors. This includes the translator, the illustrator etc.
Genres	String	The category of books under which the book can be classified such as Horror, Comedy etc. A single book can be classified under multiple genres.
Publishing Date	String	Date of publishing.
Pages	Integer	Number of pages in the book.
Publisher	String	Name of the publisher(s) as listed in the source.
Language	String	Language.
ISBN	Integer	Unique identifier information (International Standard Book Number).
ISBN13	Integer	Unique identifier information (International Standard Book Number).
URL	String	URL of the HTML page from where the data was scraped.

Information about the tables

CSV Filename	Web Source	Number of tuples
bookdepository.csv	https://www.bookdepository.com	3968
goodreads.csv	https://goodreads.com	3794

Open Source tools used.

- Scrapy - Scrapy is a web crawling framework, written in Python. It is a general purpose web crawler and can also be used for web scraping and extracting data using APIs. Using Scrapy, we define spiders for each web source. And Scrapy provides the framework for processing the response for each request fired. Also, support for crawling the multiple links in the HTML is provided by Scrapy.
- The lxml XML toolkit (used internally by Scrapy) is a Pythonic binding for the C libraries libxml2 and libxslt. It is unique in that it combines the speed and XML feature completeness of these libraries with the simplicity of a native Python API, mostly compatible but superior to the well-known ElementTree API