**CRAWLING & EXTRACTING STRUCTURED DATA FROM WEBPAGES DATA SCIENCE – CS838**  
**PROJECT STAGE II**

**Adarsh Kumar Varun Batra Vibhor Goel**

[kumar92@wisc.edu](mailto:kumar92@wisc.edu) [vbatra@wisc.edu](mailto:vbatra@wisc.edu) [vgoel5@wisc.edu](mailto:vgoel5@wisc.edu)

Web Sources:  
We chose to extract books and relevant information about them. For the purpose of extraction of data, we chose the below two sources:

***Amazon*** : <https://www.amazon.com/>Amazon started as an online bookstore nearly 20 years back. Currently, apart from being the biggest online marketplace of various kinds of goods, it also sells books. It helps you explore Earth's Biggest Bookstore where we can find current paperback books, Kindle eBooks and Audible audiobooks across various genres like Literature, Fiction, Mystery, Thrillers, Cooking, Dating, Comics, Romance, Science, Fantasy etc. They provide details about each book along with the price.

***Goodreads*** : <https://www.goodreads.com/>Goodreads is the world’s largest site for readers and book recommendations. It helps people find and share books they love. Goodreads was launched in January 2007 and provides extensive details about a book.

Entity Extracted:

We extracted books of different categories like mystery, love, dating, religion, etc. from both the data sources. This helped us ensure overlaps between entries in the 2 extracted csv tables.

Extraction Methodology and Open Source Tools used:

We made use of the DOM structure of HTML to extract data. We used Python and it’s library BeautifulSoup to extract the various properties of the books from the HTML DOM structure. [Beautiful Soup](http://www.crummy.com/software/BeautifulSoup/) is a Python library for pulling data out of HTML and XML files. It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. It commonly saves programmers hours or days of work. We crawled across multiple pages and extracted book information using the above-mentioned tool. Creating a DOM based crawler fir crawling Goodreads was relatively easy as their DOM structure was pretty straight forward. Amazon crawler required some fine tuning of the <div> “class” names to extract the information of use.

Results:

Number of tuples for Amazon in amazon.csv =   
Number of tuples for Goodreads in goodreads.csv = 3101

Schema:  
The schema for the 2 tables is described bellow

|  |  |  |
| --- | --- | --- |
| Attribute | Datatype | Description |
| ID | Integer | Primary Key |
| Name | String | Book name |
| Author | String | Writer of the book |
| Rating | Float | User Rating of the book out of 5 stars |
| Format | String | The type of book - Kindle/Paperback/Kindle Edition |
| Year | Integer | The year of book launch |