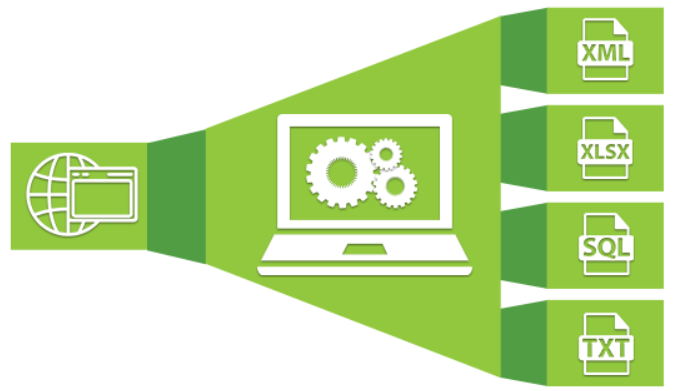
Documentation on Data Scrapping Part

Research Paper Engine



Name: Nivitus

Role: Data Science

-Intern

Date: 14/09/2020

**Introduction:**

Data scraping is a technique in which a computer program extracts data from human-readable output coming from another program. In this project part I wrote a code for scrapping the data from three kinds of research paper websites. What are the types of tools and packages I used we’ll see in upcoming part.

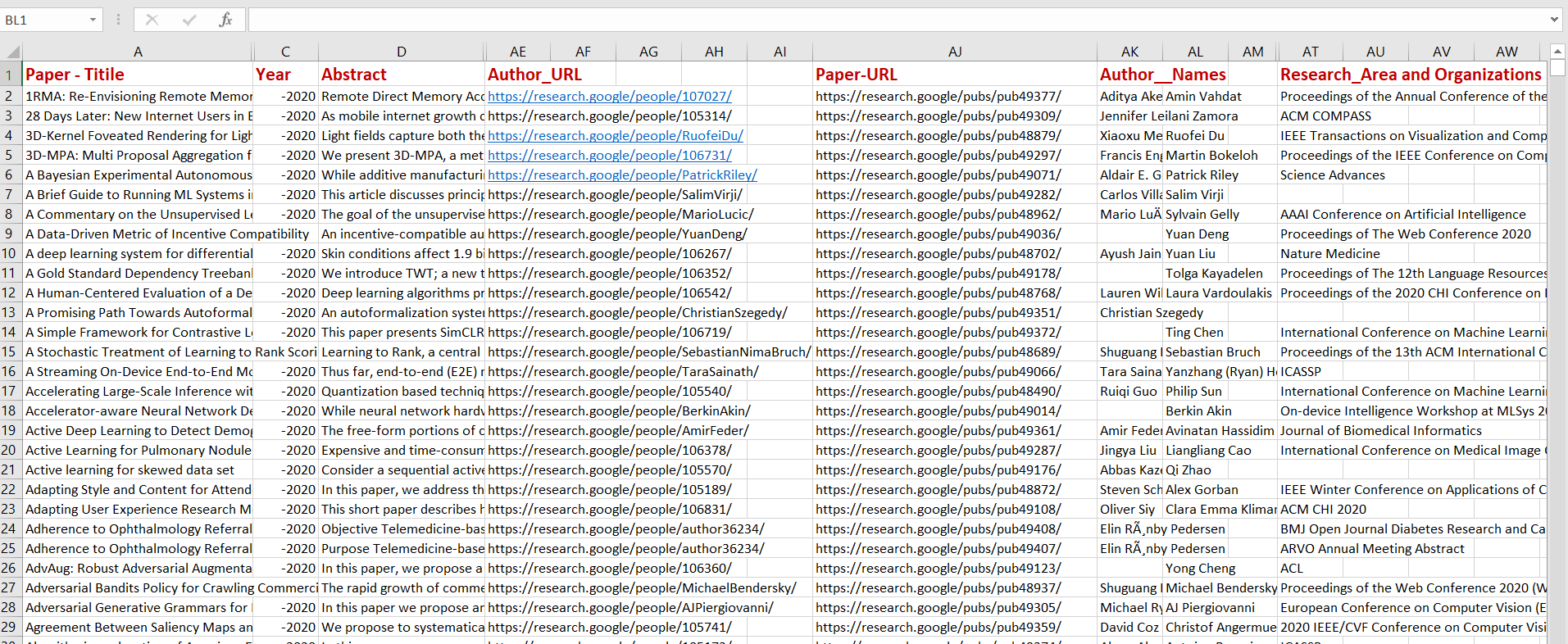
**Google Scholars:**

Google Scholar is a freely accessible web search engine that indexes the full text or metadata of scholarly literature across an array of publishing formats and disciplines. Here what I’m done I just used a tool that’s called Instant Scraper and Scrapy that was the tools for helped me to scraped the data from google scholars. Here I’m not using any python code for scrape the data from google scholars.

**Data Scraped from Google Scholars:**

* Paper Title
* Paper Title URL
* Posted Year
* Abstract
* Author Name
* Author Address
* Author URL
* Paper Publication Location
* Research Area

**Example Data Scraped Screen Shot:**



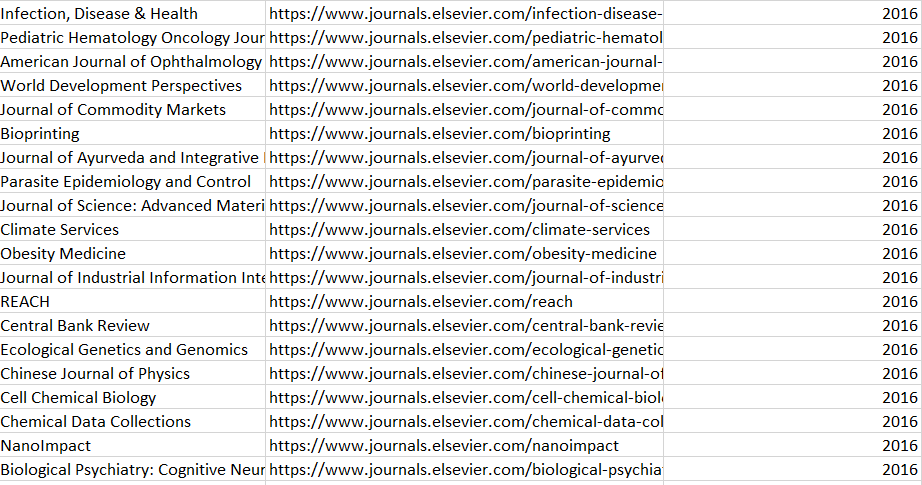
**Elsevier:**

Here what I’m done I just used a selenium and beautifulsoup4 libraries within a help of scrapped the data from Elsevier Website. I just used python code for scrapped the data with help of chrome Driver for accessing Elsevier to scrape the data automatically.

**Data Scraped from Elsevier:**

* Paper Title
* Paper Title URL
* Posted Year
* Abstract
* Author Name
* Author URL
* Research Area

**Example Data Scraped Screen Shot:**



**Python Code for Scrape the Data from Elsevier:**

from selenium import webdriver

from selenium.webdriver.common.by import By

from selenium.webdriver.chrome.options import Options

from selenium.webdriver import chrome

import pandas as pd

import os

import time

path = '/home/nivitus/WebScrap/chromedriver'

chrome\_options = Options()

chrome\_options.add\_argument("-- incognito")

browser = webdriver.Chrome(path, options=chrome\_options)

pages = 2

url = "https://www.elsevier.com/catalog?producttype=journal"

def getdata(start\_url,pgs):

current = 1

urls = browser.get(start\_url)

data = {}

df = pd.DataFrame(columns=['Paper\_Title','Paper\_URL','Abstract','Author\_Name','Author\_URL','Research\_Area','Author\_Address'])

while current < pages:

books = browser.find\_elements\_by\_xpath('//\*[@id="maincontent"]/section[1]/div[2]')

for book in books:

for b in book.find\_elements\_by\_xpath('//\*[@id="maincontent"]/section[1]/div[2]/div[1]'):

data['Paper\_Title'] = b.find\_elements\_by\_xpath('//\*[@id="maincontent"]/section[1]/div[2]/div[1]/div[1]/div/div[2]/h5/a/text()')[0].text

data['Paper\_URL'] = b.find\_elements\_by\_xpath('//\*[@id="maincontent"]/section[1]/div[2]/div[1]/div[1]/div/div[2]/h5/a')[0].text

data['Post\_Year'] = b.find\_elements\_by\_xpath('//\*[@id="maincontent"]/section[1]/div[2]/div[1]/div[1]/div/div[2]/div/p[2]/text()')[0].text

for a in books.find\_elements\_by\_xpath('//\*[@id="maincontent"]/section[1]/div[2]/div[1]/div[2]/div/a'):

data['Abstract'] = a.find\_elements\_by\_xpath('//\*[@id="PublicationDescription"]/div[2]/p[1]/text()')[0].text

data['Author\_Name'] = a.find\_elements\_by\_xpath('//\*[@id="Title"]/div[2]/div/div[3]/span[2]')[0].text

data['Author\_URL'] = a.find\_elements\_by\_xpath('//\*[@id="Title"]/div[2]/div/div[3]/a')[0].text

data['Research\_Area'] = a.find\_elements\_by\_xpath('//\*[@id="PublicationDescription"]/div[2]/p[3]/em[2]/a')[0].text

for add in books.find\_elements\_by\_xpath('//\*[@id="Title"]/div[2]/div/div[3]/a'):

data['Author\_Address'] = add.find\_elements\_by\_xpath('//\*[@id="Content1"]/div[1]/div[2]/span[2]/text()')[0].text

df = df.append(data, ignore\_index=True)

current += 1

next = browser.find\_elements\_by\_xpath('//\*[@id="content"]/div/div[3]/div/div/div/div[1]/div[2]/ul/li[7]/a')[0].click()

return df

getdata(url, pages).to\_excel('/home/nivitus/Elsevier.xls')

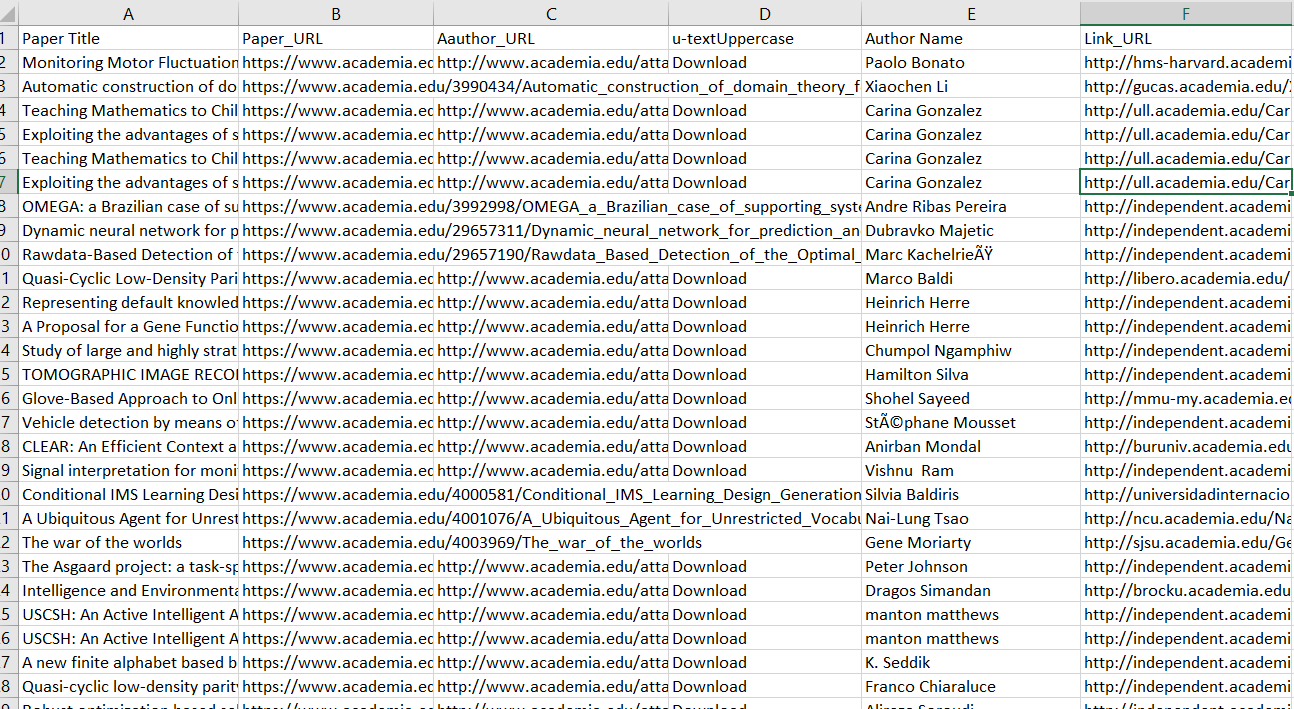
**Academia.edu**

Here what I’m done I just used a selenium and beautifulsoup4 libraries within a help of scrapped the data from Academia Website. I just used python code for scrapped the data with help of chrome Driver for accessing Academia to scrape the data automatically.

**Data Scraped from Academia.edu:**

* Paper Title
* Paper Title URL
* Posted Year
* Abstract
* Author Name
* Author Address
* Author URL
* Research Area

**Example Data Scraped Screen Shot:**



**Python Code for Scrape the Data from Academia.edu:**

from selenium import webdriver

from selenium.webdriver.common.by import By

from selenium.webdriver.chrome.options import Options

from selenium.webdriver import chrome

import pandas as pd

import os

import time

path = '/home/nivitus/WebScrap/chromedriver'

chrome\_options = Options()

chrome\_options.add\_argument("-- incognito")

browser = webdriver.Chrome(path, options=chrome\_options)

pages = 2

url = "http://www.academia.edu/Documents/in/Artificial\_Intelligence?page=1"

def getdata(start\_url,pgs):

current = 1

urls = browser.get(start\_url)

data = {}

df = pd.DataFrame(columns=['Paper\_Title','Paper\_URL','Abstract','Author\_Name','Author\_URL','Research\_Area'])

while current < pages:

books = browser.find\_elements\_by\_xpath('//\*[@id="content"]/div/div[3]/div/div/div/div[1]/div[1]')

for book in books:

for b in book.find\_elements\_by\_xpath('//\*[@id="content"]/div/div[3]/div/div/div/div[1]/div[1]/div[1]'):

data['Paper\_Title'] = b.find\_elements\_by\_xpath('//\*[@id="content"]/div/div[3]/div/div/div/div[1]/div[1]/div[1]/div/div[1]/div')[0].text

data['Paper\_URL'] = b.find\_elements\_by\_xpath('//\*[@id="content"]/div/div[3]/div/div/div/div[1]/div[1]/div[1]/div/div[1]/div/a')[0].text

data['Post\_Year'] = b.find\_elements\_by\_xpath('//\*[@id="content"]/div/div[3]/div/div/div/div[1]/div[1]/div[1]/div/ul/li[3]/ul/script')[0].text

data['Abstract'] = b.find\_elements\_by\_xpath('//\*[@id="content"]/div/div[3]/div/div/div/div[1]/div[1]/div[2]/div/div[2]/div/div[2]')[0].text

data['Author\_Name'] = b.find\_elements\_by\_xpath('//\*[@id="content"]/div/div[3]/div/div/div/div[1]/div[1]/div[2]/div/ul/li[3]/ul/li[1]/span/span/a')[0].text

data['Author\_URL'] = b.find\_elements\_by\_xpath('//\*[@id="content"]/div/div[3]/div/div/div/div[1]/div[1]/div[2]/div/ul/li[3]/ul/li[1]/span/span/a')[0].text

data['Research\_Area'] = b.find\_elements\_by\_xpath('//\*[@id="content"]/div/div[3]/div/div/div/div[1]/div[1]/div[1]/div/ul/li[3]/ul/li[5]/span/a[1]')[0].text

df = df.append(data, ignore\_index=True)

current += 1

next = browser.find\_elements\_by\_xpath('//\*[@id="content"]/div/div[3]/div/div/div/div[1]/div[2]/ul/li[7]/a')[0].click()

return df

getdata(url, pages).to\_excel('/home/nivitus/AI\_Data.xls')

**Conclusion:**

Here I have done the web scrapping part for our Project from Google Scholars, Elsevier and Academia. Some of data scrapped used by tools some of that used python code and Data scrapping libraries in python.