#Project:

To develop a specialized R program to crawl, parse and extract all articles published in a specific journal.

#Group 0: Aishwarya Rasal and Mounica Sreedhara

Readme.txt

How to run?

Copy all the R scripts in the wd (working directory) - Execute Group0.R only

Group0.R

# group\_id was generated using above steps and the group id generated was 0 (G3 Journal)

Aishwarya = utf8ToInt("Rasal")

Mounica = utf8ToInt("Sreedhara")

group\_id = (sum(Aishwarya) + sum(Mounica) )%%10

Step1: Extract main page url and extract total page information store it in main\_page.html.

Step2: Extract article page url from the main page and extract total page information store it in different files as number of article webpages crawled.

Step3: Save all article page with \*.html extension.

Step4: Extract all the fields that occur in every article page. DOI, Title, Authors, Author Affiliation, Corresponding Author, Corresponding Author’s Email, Publication Date, Abstract are extracted and stored in a data frame.

Step5 – Write the data into a text file called G3journals.txt

Step6 – Read the G3.journals.txt file in R and store it as a dataframe , named reading.output, with DOI, Title, Authors, Author Affiliation, Corresponding Author, Corresponding Author’s Email, Publication Date Abstract, Keywords and FullText as column names. The Keywords are marked as NA as they do not occur in my article pages.

Step7 – Provide the summary of the fields in a excel sheet of all the webpage crawled. Write the dataframe data onto a excel sheet named, G3Journals\_summary\_of\_fields.xlsx.

Library Used:

library(bitops)

library(RCurl)

library(XML)

library(xml2)

library(httr)

library(stringr)

library(Rcrawler)

library(rvest)

library(rlist)

library(openxlsx)

Function Used:

LinkExtractor() – to extract articles url from webpages

Grep()- search for matches to argument pattern within each element of a character vector

List.append()- to append a data to a existing list.

Read\_html()- Read in the content from a .html file.

xpathSApply()-xpathSApply is a version of xpathApply which attempts to simplify the result if it can be converted to a vector or matrix rather than left as a list. It uses XPath of the html pages to get required content of the page.

htmlParse()-Parses an XML or HTML file or string containing XML/HTML content, and generates an R structure representing the XML/HTML tree

Gsub()-perform replacement of the first and all matches respectively.

Trimws()-trims white spaces in a string. It can be leading or trailing spaces or both.