

Web Scraping do ResearchID: proposta de sistema para o monitoramento de Índice H de pesquisadores no Brasil

Alexandre Ribas Semeler^a, Adilson Luiz Pinto^b e Arthur Longoni Oliveira^c

Resumo: A dimensão dos dados aponta uma nova tendência de estudos e práticas que vem sendo adotada por cientistas da informação interessados em estudos métricos que visem o uso de dados de citação e referências. Essas abordagens ressaltam métodos e tecnologias que garantem a interoperabilidade e a criação de uma identidade única para autores e de documentos científicos. O ResearcherID é um sistema de identificação de autores científicos, criado em 2008 pela Thomson Reuters com o objetivo de resolver o problema da identificação de autores. O ResearcherID foi adotado pelo Conselho Nacional de Pesquisa do Brasil como fonte de dados para coleta do Índice H nacional. Este indicador é aplicado para medir a produtividade e visualização do impacto de cientistas baseando-se nos seus artigos mais citados. Nesse contexto, a proposta desse estudo será a de medir o índice H dos pesquisadores brasileiros cadastrados no ResearcherID. Os procedimentos metodológicos exigem a aplicação de conhecimentos inerentes a linguagem de programação Python. São utilizadas técnicas de web scraping para recuperação de informação no ResearcherID. O resultado do estudo consolida-se na forma de dois scripts em Python (Anexo A e B) tais programas objetivam-se como base para a elaboração de um sistema de monitoramento do índice H no ResearcherID.

Palavras-chave: Dados de Citação; ResearcherID; Índice H; Web Scraping.

Web Scraping in ResearchID: proposal to the monitoring system of H-index of the researchers in Brazil

Abstract: The data dimension presents a new trends in studies and practices adopted by information scientists interested in metric studies that aim at the use of data citation and data references. These trends emphasize methods and technologies that ensure interoperability and the creation of a unique identity to authors and scientific documents. The ResearcherID is a system of identification of scientific authors, created in 2008 by Thomson Reuters with the purpose of solving the problem of the identification of authors. The ResearcherID was adopted by the National Research Council of Brazil as data source to validation and collection of the national H-index. This indicator is applied to measure the productivity and visualization of the impact of scientists based on their most cited articles. The aim of this study will be to measure the H-index of Brazilian researchers enrolled in ResearcherID. The methodological procedures require the application of knowledge inherent in the Python programming language. The web scraping techniques are used for information retrieval in ResearcherID. The results of the study are consolidated in two Python scripts (Annex A and B). Such programs are intended as a basis for the development of an H-index monitoring system in ResearcherID.

Keyword: Data Citation; ResearcherID; H-index; Web Scraping.

-
- a Universidade Federal do Rio Grande do Sul (UFRGS). E-mail: alexandre.semeler@ufrgs.br. ORCID: <https://orcid.org/0000-0002-8036-4271>. Currículo: <http://lattes.cnpq.br/1564154234554393>
 - b Universidade Federal de Santa Catarina (UFSC). E-mail: adilson.pinto@ufsc.br. ORCID: <https://orcid.org/0000-0002-4142-2061>. Currículo: <http://lattes.cnpq.br/4767432940301118>
 - c Universidade Federal do Rio Grande do Sul (UFRGS). E-mail: arthur.holiver@gmail.com. Currículo: <http://lattes.cnpq.br/7483194676718999>

1 Introdução

A dimensão dos dados digitais aponta uma nova tendência de estudos e práticas que vem sendo adotada por cientistas da informação interessados em estudos métricos, em específico aqueles que visem o uso de dados de citação bibliográfica.

Os estudos métricos com base neste tipo de dados não são algo novo, desde a década de 1963 são desenvolvidos pelo Institute for Scientific Information vide o exemplo do Science Citation Index. No entanto, a identificação precisa de autores e suas citações junto com a integração de suas publicações em uma única identidade científica são questões ligadas ao contexto digital e justificam a realização deste estudo, que tem como objetivo identificar a variação do Índice H Brasileiro.

Segundo Hirsch (2005), o Índice H é um indicador de quantificação para a produtividade e visualização do impacto de cientistas baseando-se nos seus artigos mais citados.

O processo de busca por informação em dados de citação, principalmente, quando estes dados tratam o Índice H podem indicar uma frente de pesquisa. Uma frente de pesquisa é composta pelo conjunto de autores mais citados em um determinado campo e indicam a elite de autores. Assim, os dados de citação podem refletir a produtividade e a visibilidade dos autores.

Tendo em vista o desenvolvimento dos índices de citação e algumas tecnologias que retratam o Índice H mundial, conforme explanado por Rice e Southall (2016), a Clarivate Analytics desenvolveu em 2008 ResearcherID sistema utilizado para rastrear citações em conjuntos de dados da literatura científica de impacto internacional.

O ResearcherID é um sistema de identificação de autores científicos, criado pela Thompson Reuters com o objetivo de resolver o problema da identificação de autores. O ResearcherID se integra com a Web of Science (Science Citation Index), é um identificador digital único, gratuito e persistente, que distingue um pesquisador de outro e resolve o problema da ambiguidade e semelhança de nomes de autores e indivíduos, substituindo as variações de nome por um único código numérico facilitando o registro de informações ele automatiza a atualização das publicações e dados de citação possibilitando a interoperabilidade entre as publicações e suas citações a partir de uma única conta. (RESEARCHERID, 2018)

Esse ID científico é um código de 11 dígitos, ex.: (J-9183-2016), na forma de caracteres alfanuméricos sendo único para cada indivíduo. O ResearcherID impede ambiguidades na identificação de autores e colaboradores em publicações. Seu funcionamento é semelhante ao do Digital Object Identifier (DOI), código para objetos como artigos científicos, teses e dissertações (RESEARCHERID, 2018; RICE; SOUTHALL, 2016).

O ResearcherID é um identificador único que permite que os pesquisadores gerenciem suas listas de publicações, rastreiem suas contagens de citações e índice h. (RESEARCHERID; 2018)

A utilização desses IDs favorece a localização e a contagem das publicações, além de reunir os dados de citações recebidas pelo conjunto de trabalhos de sua autoria possibilitando a visualização do Índice H.

O conselho Nacional de Pesquisa do Brasil (CNPq) adotou o ResearcherID para validar os dados sobre citações a publicações científicas de impacto internacional na sua plataforma de currículos Lattes, vide apêndice F. O CNPq utiliza o ResearcherID, para coletar e validar

automaticamente os dados sobre o número total de citações, número de trabalhos e Índice H, dos pesquisadores que possuem publicações científicas de impacto internacional na Web of Science por meio do ResearcherID.

Nesse sentido, a proposta desse estudo será a de medir o índice H dos pesquisadores brasileiros cadastrados no ResearcherID, contar o número de publicações e visualizar o número de publicações com citação na WOS.

No que segue essa pesquisa exigiu a aplicação de conhecimentos inerentes a técnicas e tecnologias utilizadas para coleta, de dados de citação. Como método de coleta utilizou-se um conjunto de rotinas e padrões de programação para extrair as informações do ResearcherID. Tais procedimentos metodológicos exigem o desenvolvimento de máquinas de busca com a linguagem de programação Python.

A seguir apresentam-se os objetivos, os procedimentos metodológicos e os resultados desse estudo com maiores detalhes.

2 Objetivos

O objetivo geral desta proposta é o de desenvolver scripts em Python para monitorar o Índice H de pesquisadores brasileiros no ResearcherID. Para isso, desenvolvem-se os seguintes objetivos específicos:

- A) Automatizar a coleta do Índice H de pesquisadores registrados no ResearcherID;
- B) Criar scripts de web scraping em linguagem de programação Python para coleta de dados de citação;
- C) Medir o índice H dos pesquisadores brasileiros cadastrados no ResearcherID, contar o número de publicações e visualizar o número de publicações com citação na WOS.

3 Procedimentos Metodológicos

Os procedimentos metodológicos fundamentam-se na web scraping das métricas de citação do ResearcherID. O método adotado para este estudo foi o web scraping em Python, método que se define como processo de extração e combinação de conteúdos web de forma sistemática e automatizada. Em tal processo, um agente de software, o scraping, simula o comportamento de navegação humano em servidores web copiando e reorganizando dados desorganizados em dados organizados (GLEZ-PENA et al., 2013).

Existem diferentes técnicas de busca para web scraping: a manipulação de HTTP, a mineração de dados, as ferramentas de scraping, a cópia manual e os microformatos. A manipulação de HTTP permite a colheita de dados estáticos e dinâmicos de um site através de uma solicitação HTTP. A mineração de dados é um processo automático que reconhece as informações de um site de acordo com scripts predefinidos que contêm dados incorporados. Ferramentas de scraping são softwares utilizados para extrair informações relacionadas a sites ou a funcionalidades e/ou estruturas de dados na web. As ferramentas de scraping também servem para extrair dados de redes sociais e são úteis para todos os tipos de atividades de marketing web. A cópia manual, embora existam ferramentas de scraping disponíveis, às vezes é necessária, por exemplo, em alguns casos em que a informação do site for bloqueada contra qualquer forma web

scraping. Por fim, os microformatos são referentes às tecnologias da web semântica, como dados abertos vinculados via RDF. Esses são conjuntos de informações geralmente elaboradas para serem intercambiadas via vocabulários e ontologias (WEBSTER, 2015).

O funcionamento de um web scraping ocorre da seguinte maneira: (a) acesso ao site via protocolo de comunicação web (HTTP), durante o processo de solicitação e resposta entre um cliente, normalmente um navegador da web, e um servidor da web; (b) o programa de scraper analisa o HTML e pode extrair seu conteúdo; (c) armazena o output em conteúdo textual, com o objetivo de transformar o conteúdo extraído em uma representação estruturada para posterior análise e armazenamento (GLEZ-PENA et al., 2013).

O objetivo da web scraping é transformar os dados desestruturados da web em representações estruturadas, como aquelas de formatos tabulares, por ex.: os tipos de arquivos Comma Separated Values (CSV), Tab Separated Values (TSV), e/ou eXtensible Markup Language (XML). Esses formatos permitem a análise do conteúdo de sites da mesma forma que permitem salvar a navegação feita os browsers web, como o Chrome ou FireFox Mozilla.

No que segue apresenta-se uma summarização dos procedimentos metodológicos, conforme o quadro 1.

Quadro 1 – Caracterização do método.

Caracterização do método	
Tipo de pesquisa	Exploratória e Descritiva
Estratégia	Uso de linguagens de programação para coletar o esquema de métricas do ResearcherID
Natureza	Quantitativa
Panorama geral da investigação	Índice H de pesquisadores brasileiros
Amostra	43.114 – IDS em 17 set. 2018
Corpus total	108.000 – IDS em 17 set. 2018
Instrumentos de coleta	Web scraping
Softwares	IDE = Pycharm Python=2.7 Módulos = Codecs, BeautifulSoup, Selenium
Fonte de coleta de dados	ResearcherID

Fonte: Dados da Pesquisa (2018).

Com base nesses procedimentos apresentam-se os resultados da pesquisa.

4 Resultados

Os resultados obtidos nesse estudo representam um possível modelo para execução automatizada do processo de coleta de dados de citação oferecidos pelo ResearcherID. Subdividem-se em um plano técnico, criação de scripts para extração automática das informações no ResearcherID e a interpretação destes dados.

4.1 Resultados técnicos

Como resultado técnico obteve-se dois scripts em Python os quais são aplicados para o monitoramento do índice H dos pesquisadores cadastrados no ResearcherID, contar o número de publicações e visualizar o número de publicações com citação desses pesquisadores na Web of Science. Os mesmos podem ser visualizados no Anexo A e B deste trabalho.

O apêndice A, apresenta o script1, o qual executa a primeira coleta de dados no ResearcherID. O script1 automatiza o web scraping das informações sobre: o nome de pesquisadores, sua instituição, o código ID e as áreas do conhecimento relacionadas ao seu ID. O script1 salva uma lista contendo, o um conjunto de dados com todos os IDs de pesquisadores brasileiros que possuem registro no ResearcherID, a lista contém (43.114) pesquisadores brasileiros registrados em 19 de novembro de 2018. O script1 possui um dispositivo de tratamento de falhas, que garante que quando interrompida a extração, em caso de timeout na página, a extração é reiniciada no último ponto válido. O script1 também evita as duplicações de IDs.

Os resultados da coleta feita pelo script1 são salvos em um arquivo .TSV e contem as seguintes informações: Nome do pesquisador, a instituição, o código no ResearcherID e as palavras-chave. Estes dados servem para o script2 como base para uma nova coleta de dados sobre as métricas de citação do ResearcherID .

O apêndice B representa o script2, o qual lê a lista de dados escrita pelo script1 e executa a coleta de informações sobre as métricas de citação do ResearcherID. O script2 lê os registros do arquivo .TSV e gera links (<http://www.researcherid.com/rid/+ ID do pesquisador>) usando os IDs do arquivo .TSV. Estes Links são únicos e representam os dados de citação de cada pesquisador. O script2 é paralelizado em 8 processos iguais, cada processo de coleta de informações de (1/8) dos links gerados a partir do arquivo .TSV, essas informações serão salvadas em um único arquivo .TSV contendo os dados de citação da base, em especial: ID, total de publicações na lista, publicações com dados de citação, A soma de citações no tempo, a média de citações por artigo e o Índice H dos pesquisadores brasileiros.

4.2 Interpretações da coleta

Os resultados da coleta permitem não apenas a visualização do Índice H, mas também outras informações apresentadas pelo ResearcherID como: total de publicações na lista, publicações com dados de citação, a soma de citações no tempo, a média de citações por artigo e o Índice H. Ressalta-se que os Índices H discrepantes e aqueles que não correspondem a autores únicos, como aqueles utilizados para medir instituições ou mesmo aqueles feitos erroneamente foram excluídos da análise por não representarem o objetivo deste trabalho.

Assim a coleta permitiu visualizar algumas situações, conforme o apêndice C, de um total de 40.114 pesquisadores cadastrados no ResearcherID, (9560) pesquisadores não possuem métricas no ResearcherID (N/A), ou seja nunca foram citados no Science Citation Index. Outros (960) possuem publicações com zero citação, ou seja, estes trabalhos não possuem impacto na WOS ou ainda não foram citados. As 10 maiores frequências de índice a H estão entre 1 e 10: (1=4006, 2=3454,3=3244, 4=2752, 5=2572, 6=2141, 7=1823, 8=1651, 9=1410, 10=1231).

Com base nos 40114 IDs a média do Índice H nacional é (7,97), levando-se em consideração pesquisadores únicos e aqueles que possuem índice H maior que zero, ou seja, um montante de (32.255) pesquisadores brasileiros.

A frente de pesquisa nacional pode ser visualizada no apêndice D e concentra-se no campo científico da Física. Conforme a Tabela 2 contida neste anexo o ID (B-2946-2012) possui o maior índice H nacional válido (96). Este índice é de um professor do Instituto de Física Gleb Wataghin da Universidade Estadual de Campinas (UNICAMP). O segundo maior índice H (92) também corresponde a área de Física, conforme o ID (L-6239-2016), pertence a um professor do Instituto de Física da Universidade de São Paulo. Ambos atuam na área de Física Nuclear e são Bolsistas de Produtividade em Pesquisa do CNPq – Nível 1D.

O apêndice D também permite visualizar e ranquear os pesquisadores brasileiros conforme a variação dos Índices H mais altos (mínimo 60 e máximo 96). O ID que recebeu mais citações (39544) é (L-1621-2016) de um professor da Universidade Estadual do Rio de Janeiro que também atua na área de Física nuclear sendo Bolsista de Produtividade em Pesquisa do CNPq - Nível 1C.

Já as publicações com citação na WOS compreendem o montante de (877.624) estas são publicações que receberam no mínimo 1 citação em periódico com fator de impacto. O número total de publicações registradas nas listas foi (1.146.855), no entanto este número é prejudicado ao representar o número de produções do Brasil na WOS, pois alguns pesquisadores não preencheram seus IDs apenas com a sua produção na WOS mas com outras fontes de dados de citação como Google Scholar e Scopus.

Por fim, o ID (K-5168-2013) possui a maior média de citações recebidas (99.88) no ResearcherID a coordenadora do Núcleo de Pesquisa em Asma e Inflamação das Vias Aéreas (NUPAIVA), com índice H (22) a pesquisadora da Universidade Federal de Santa Catarina valida todasua produção na Plataforma Lattes.

No entanto, chamou a atenção que os maiores índices H brasileiros (140), (105) e (103) não serem válidos, pois foram realizados com objetivos diversos e não com o objetivo de validar os dados de citação no Lattes CNPq. É necessário ter em mente que o ResearcherID é um identificador pessoal de autor e possui o objetivo de fornecer métricas para produção de autores únicos e não de instituições, ressaltando também que não é uma ferramenta para pesquisa bibliográfica.

Assim o índice H 140, (A-9780-2017), atribuído ao Instituto de Física, Gleb Wataghin da UNICAMP e o índice H 105, (M-2664-2016), que representa o Instituto de Física de São Carlos da Universidade de São Paulo prejudicam a visualização do índice H nacional de autores já que são índices de um conjunto de autores de uma instituição. Outra discrepância é o índice H 103 (E-4724-2015), de um estudante do Laboratório de Raioisótopos do Centro de Energia Nuclear na Agricultura – CENA, que foi construído de maneira errônea pois o autor em sua lista pessoal de publicações insere produções científicas de outros autores de certo ele utilizou a plataforma para visualizar a produção em sua área de pesquisa desconhecendo o objetivo do ResearcherID, tão pouco ele valida seu Lattes com os dados.

Este três casos ressaltam a relevância desta proposta de trabalho. Pois a visualização do Índice H nacional é usada pelo CNPq como estratégia para fomentar e desenvolver novas pesquisas, pautando-se na relevância e no impacto dos pesquisadores atribuída pela WOS, nas mais diversas áreas do conhecimento. No entanto, é necessário fazer o uso correto da plataforma ResearcherID que é um instrumento de validação de citações utilizada pela plataforma Lattes do CNPq.

5 Considerações Finais

O Índice H é usado pelo CNPq como um dos indicadores para pautar a distribuição de Bolsas de Pesquisa e para pautar a relevância acadêmica da produção científica no Brasil. Conhecer o Índice H nacional torna-se tarefa de suma relevância. Pois o Índice H indica o impacto e a relevância dos trabalhos mais citados ao longo do tempo. Revelando a visibilidade e o uso da produção científica de impacto internacional de cada pesquisador.

Considera-se esse estudo como uma estratégia fundamental para medir a variação do Índice H nacional, a variação da média de citações de impacto internacional, medir a soma das citações da produção científica nacional, assim como para visualizar o total de publicações nacionais na lista do ResearcherID com citações.

Por fim, ressalta-se que os dados de citação e os IDs de autores são ferramentas necessárias para visualização do impacto internacional do que é produzido em nosso país, sendo automatização desse processo de suma aplicabilidade em estudos métricos que visem o uso de dados para qualificar a produção científica nacional. Conclui-se que os usuários brasileiros deveriam ter maior critério ao utilizar o ResearcherId, pois a ferramenta é utilizada pelo CNPq para validar o impacto da produção nacional na Web of Science uma das principais e mais respeitadas bases de dados e documentos bibliográficos.

Agradecimentos

“O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) - Código de Financiamento 001”.

Referências

GLEZ-PEÑA, D. et al.. Web scraping technologies in an API world. **Briefings in Bioinformatics**, v. 15, n. 5, p. 788-797, 2013. Disponível em:<<http://bib.oxfordjournals.org/content/15/5/788>>. Acesso em: set. 2018.

HEIRICH, J. An index to quantify an individual's scientific research output. **PNAS**, v. 102, n. (46), 2005. Disponível em: <https://doi.org/10.1073/pnas.0507655102>. Acesso em: set. 2018.

PYTHON. Disponível em: <<https://www.python.org/>>. Acesso em set. 2018.

ResearcherID. Disponível em: <<http://www.researcherid.com>> Acesso em: set. 2018.

RICE, R.; SOUTHALL, S. **The data librarian's handbook**. London: Facet Publishing, 2016.

WEBSTER, S. **What Is Scraping?** The Basics For Everyone. 2015. Disponível em: <https://myhelpster.com/what-is-scraping-the-basics-for-everyone/>. Acesso em: set. 2018.

Apêndice A – Script1: coleta de informações gerais no ResearcherID

```

# coding: utf-8
"""
Created on Sep 17 2018
"""

import codecs
from bs4 import BeautifulSoup
import selenium.common.exceptions
# from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as ec
from selenium.webdriver import Firefox
# import os

url = "http://www.researcherid.com/ViewProfileSearch.action"    # define url alvo
# Caso a execução seja feita com Chrome
# chromedriver = "../scripts/chromedriver"                      # path do chromedriver
# os.environ["webdriver.chrome.driver"] = chromedriver          # abre browser
driver = webdriver.Chrome(chromedriver)

# abre arquivo para escrita dos resultados
with codecs.open('../dados/csv/RESEARCHID_Extraction2.tsv', 'w', 'utf-8-sig') as results:
    results.write('Name\tInstitution\tResearcher ID\tKeywords\n')
    # escreve headers
results.close()
id_list = []
counter = 0      # inicializa contadores de resultados e erros
errors = 0

driver = Firefox()
while True:
    try:
        driver.get(url)                                     # navega para a url
        # espera carregamento da combobox para escolher o País alvo
        WebDriverWait(driver, 10).until(ec.presence_of_element_located((By.CSS_SELECTOR,
            '#country > option:nth-child(32)'))).click()       # country
        break
    except selenium.common.exceptions.TimeoutException:
        continue

# clica para submeter pesquisa
driver.find_element_by_css_selector('#submitImage > img').click()
# define limite de páginas para serem acessadas nos resultados
page_limit = int(WebDriverWait(driver, 10).until(ec.presence_of_element_located(
    (By.CSS_SELECTOR, '#criteria_requestedPage_total'))).text) // 50 + 1
# espera combobox carregar para mostrar 50 resultados por página
WebDriverWait(driver, 10).until(ec.presence_of_element_located((By.CSS_SELECTOR,
    '#resultsPerPage > option:nth-child(3)'))).click()

for page in range(page_limit):    # percorre páginas de resultados da busca
    # espera tabela de resultados ser carregada
    WebDriverWait(driver, 10).until(ec.presence_of_element_located(
        (By.CSS_SELECTOR, '#resultContainer > table:nth-child(23) > tbody > tr:nth-
        child(2) > td > table > tbody')))
    # define raiz da árvore de resultados para parsing
    root = BeautifulSoup(driver.page_source, 'lxml').find('td', {'class':
        'resultsTableBox'}).findNext('tbody')
    with codecs.open('../dados/csv/RESEARCHID_Extraction2.tsv', 'a', 'utf-8-sig') as results:
        for item in root.findAll('tr')[1:]:    # percorre itens na tabela
            data = item.findAll('td')           # lista todos os resultados
            rid = data[4].getText().replace('\n', '').replace(' ', '')
            if rid in id_list:
                continue
            else:
                id_list.append(rid)
                # escreve resultados no arquivo de output
                results.write('${}\t${}\t${}\n'.format(data[1].a.getText().replace('\n', ''),
                    data[2].getText().replace('\n', ''),
                    rid,
                    data[5].getText().replace('\n', '')))
            counter += 1      # incrementa contador de resultados
        results.close()
    if page < page_limit:    # testa se as páginas de resultados já acabaram
        # clica em next page caso contrário
        WebDriverWait(driver, 10).until(ec.presence_of_element_located(
            (By.CSS_SELECTOR, '#resultContainer > table:nth-child(22) > tbody > tr >
            td:nth-child(2) > table > '
            'tbody > tr > td:nth-child(7) > span > img')).click())
        print counter
        try:    # tenta esperar pelo carregamento de um item da página como teste
            WebDriverWait(driver, 20).until(ec.presence_of_element_located(
                (By.ID, 'pageNumnull')))
        except selenium.common.exceptions.TimeoutException:    # em caso de timeout, trata o
            errors
            driver.get(url)    # navega para a url alvo novamente
            errors += 1        # incrementa counter de erros
            # reinicia a pesquisa
            WebDriverWait(driver, 30).until(ec.presence_of_element_located(
                (By.CSS_SELECTOR, '#country > option:nth-child(33)')).click())
            driver.find_element_by_css_selector('#submitImage > img').click()
            WebDriverWait(driver, 40).until(ec.presence_of_element_located(
                (By.CSS_SELECTOR, '#resultsPerPage > option:nth-child(3)')).click())
            page -= 1          # reduz contador de páginas navegadas em 1
            element = WebDriverWait(driver,
                10).until(ec.presence_of_element_located((By.XPATH, '//*[@id="pageNumnull"]')))
            element.clear()    # limpa caixa de texto com número de páginas
            element.send_keys('${}'.format(page))    # escreve número de páginas na caixa de texto

        driver.find_element_by_xpath('/html/body/div[3]/form[2]/table/tbody/tr/td/div/table/tbody/
        tr[2]/td[2]/div/div/')

        driver.find_element_by_xpath('//*[@id="pageNumnull"]').click()    # clica em "go"
        driver.quit()
        print errors, 'errors happened. Extraction finished.'

```

Fonte: Dados da pesquisa (2018).

Apêndice B – Script2: coleta das métricas de citação do ResearcherID

```

# coding: utf-8
# Created on Aug 27 2018
#####
import codecs
from bs4 import BeautifulSoup
from multiprocessing import Pool
import selenium.common.exceptions
from selenium.webdriver.common.by import By
from selenium.webdriver.common.wait import WebDriverWait
from selenium.webdriver.support import expected_conditions as ec
from selenium.webdriver.firefox.options import Options
from selenium.webdriver import Firefox
from selenium import webdriver
from selenium.webdriver.chrome.options import Options
from selenium import webdriver
#####

with codecs.open('../dados/csv/RESEARCHERID-ALL-METRICS.tsv', 'r', 'utf-8'):
    id_list = master.readlines()
    master.close()
for x in range(len(id_list)):
    id_list[x] = id_list[x].split('\n')[0]
#####
options = Options()
options.add_argument("--headless")

class Control:
    def __init__(self):
        self.counter = 0
        self.articles_accumulator = 0
        self.citation_accumulator = 0
        self.h_index_accumulator = 0
        self.existent_h_index_counter = 0

    def id_cleaning(a_link):
        if '\r\n' in a_link: # tests finais de linhas para verificar se usam \n ou \r\n
            return a_link.replace('\r', '').replace('\r\n', '\n') # troca final por \n e tira blank spaces
        else:
            return a_link.replace(' ', '').replace('\n', '\r\n')

    def get_metrics_info(a_soup):
        return (a_soup.find(id='metrics-articleCountForMetrics').getText(),
                a_soup.find(id='metrics-timesCited').getText(),
                a_soup.find(id='metrics-averagePerItem').getText(),
                a_soup.find(id='metrics_hindex').getText())

    def parallel_scraping(a_trio):
        global id_list
        driver = Firefox(firefox_options=options)
        for item in a_trio[0]: # para percorrer a lista de links
            rid = id_cleaning(item.split('\t')[2])
            if rid in id_list:
                continue
            link_name = "http://www.researcherid.com/rid/" + rid + item.split('\t')[0]
            a_trio[2].counter += 1 # increments counter para acompanhar progresso
            print a_trio[2].counter, ': Extracting: ' + link_name
            driver.get(link_name) # acessa link alvo
            try:
                driver.find_elements_by_xpath('/html/body/div[3]/div/table/tbody/tr/td[1]/table/tbody/tr[1]/td[2]/'
                                              'div/div[1]/ul[1]/li[2]/a')[0].click() # clica para mostrar citation metrics
            except IndexError:
                print 'It appears got no citation metrics' + link # exception tratada
                continue
            a_file = codecs.open(a_trio[1], 'a', 'utf-8-sig')
            try:
                WebDriverWait(driver, 10).until(ec.presence_of_element_located((By.CSS_SELECTOR, '#metrics_hindex')))
            except selenium.common.exceptions.TimeoutException:
                a_file.write(link_name[-1] + '\t' + 'TO\tTO\tTO\tTO\n')
            else:
                root = BeautifulSoup(driver.page_source, 'lxml') # parsing inicial
                data = get_metrics_info(root)
                if data[0]:
                    a_trio[2].articles_accumulator += int(data[0])
                    a_file.write(link_name[-1] + '\t' + '#\t#\t#\t#\n' + data[0])
                    if data[-1] > 0:
                        a_trio[2].citation_accumulator += int(data[1])
                        a_trio[2].h_index_accumulator += float(data[-1])
                        a_trio[2].existent_h_index_counter += 1
                else:
                    a_file.write(link_name[-1] + '\t' + '#NA\tNA\tNA\tNA\n')
            finally:
                a_file.close()
        finally:
            driver.quit() # fecha browser
        #####
    name__ = '__main__':
    pool = Pool(processes=8)
    geckodriver = '../scripts/geckodriver' # path do geckodriver para firefox
    chrome_options = Options()
    chrome_options.add_argument("--headless")
    chromedriver = '../scripts/chromedriver' # path do chromedriver
    os.environ["webdriver.chrome.driver"] = chromedriver
    with codecs.open('../dados/csv/RESEARCHID_Extraction2.tsv', 'r', 'utf-8-sig') as search:
        link_list = research.readlines()[1:] # extrai linhas do arquivo em uma lista
        research.close()
        with open('../dados/error_logs/ResearcherIDanomScraping.txt', 'w') as log:
            out_file1 = codecs.open('../dados/csv/RESEARCHIDresultsPart1-8.tsv', 'w', 'utf-8-sig') # abre arquivo de output
            out_file2 = codecs.open('../dados/csv/RESEARCHIDresultsPart2-8.tsv', 'w', 'utf-8-sig') # abre arquivo de output
            out_file3 = codecs.open('../dados/csv/RESEARCHIDresultsPart3-8.tsv', 'w', 'utf-8-sig') # abre arquivo de output
            out_file4 = codecs.open('../dados/csv/RESEARCHIDresultsPart4-8.tsv', 'w', 'utf-8-sig') # abre arquivo de output
            out_file5 = codecs.open('../dados/csv/RESEARCHIDresultsPart5-8.tsv', 'w', 'utf-8-sig') # abre arquivo de output
            out_file6 = codecs.open('../dados/csv/RESEARCHIDresultsPart6-8.tsv', 'w', 'utf-8-sig') # abre arquivo de output
            out_file7 = codecs.open('../dados/csv/RESEARCHIDresultsPart7-8.tsv', 'w', 'utf-8-sig') # abre arquivo de output
            file_list = [out_file1, out_file2, out_file3, out_file4, out_file5, out_file6, out_file7, out_file8]
            octan = len(link_list) // 8
            pool.map(parallel_scraping, [(link_list[:octan-1], # link_list[0:octan-1],
                                         file_list[0], control1),
                                         (link_list[1:2*octan-1], # link_list[1:2*octan-1],
                                         file_list[1], control2),
                                         (link_list[2*octan-1:3*octan-1], # link_list[2*octan-1:3*octan-1],
                                         file_list[2], control3),
                                         (link_list[3*octan-1:4*octan-1], # link_list[3*octan-1:4*octan-1],
                                         file_list[3], control4),
                                         (link_list[4*octan-1:5*octan-1], # link_list[4*octan-1:5*octan-1],
                                         file_list[4], control5),
                                         (link_list[5*octan-1:6*octan-1], # link_list[5*octan-1:6*octan-1],
                                         file_list[5], control6),
                                         (link_list[6*octan-1:7*octan-1], # link_list[6*octan-1:7*octan-1],
                                         file_list[6], control7),
                                         (link_list[7*octan-1:], # link_list[7*octan-1:],
                                         file_list[7], control8)])
            total_H_index = 0
            total_existent_H_index = 0
            total_articles = 0
            elements_counter = 0
            for x in control_list:
                total_H_index += x.h_index_accumulator
                total_existent_H_index += x.existent_h_index_counter
                total_articles += x.articles_accumulator
                elements_counter += x.counter
            print 'Total H index: ' + str(total_H_index)
            print 'Total existent H index: ' + str(total_existent_H_index)
            print 'Total articles: ' + str(total_articles)
            print 'Elements counter: ' + str(elements_counter)
#####

```

Fonte: Dados da pesquisa (2018).

Apêndice C – Tabela 1 – Rank de Frequências do Índice H Brasileiro

H index	Freq.	H index	Freq.
0	960	41	17
1	4006	42	25
2	3454	43	5
3	3244	44	21
4	2752	45	5
5	2572	46	12
6	2141	47	10
7	1823	48	12
8	1651	49	16
9	1410	50	7
10	1231	51	6
11	1082	52	3
12	931	53	9
13	748	54	9
14	671	55	7
15	604	56	7
16	526	57	1
17	498	58	7
18	369	59	5
19	355	60	4
20	307	61	3
21	253	62	6
22	251	63	1
23	205	67	3
24	183	68	2
25	159	69	5
26	126	71	2
27	143	73	1
28	96	75	3
29	85	76	3
30	90	77	2
31	66	80	2
32	46	83	2
33	57	84	2
34	37	85	1
35	42	86	1
36	38	92	1
37	22	96	1
38	26	103	1
39	15	105	1
40	28	140	1
		N/A	9576

Fonte: Dados da Pesquisa (2018).

Apêndice D – Frente de Pesquisa Nacional Conforme Índice H

ResearcherID	Total in Pub. List	With Citation Data	Sum of Times Cited	Average Citations/Article	H-index
B-2946-2012	438	438	31845	72.87	96
L-6239-2016	408	406	29667	73.07	92
D-3532-2012	833	820	39622	48.50	86
L-1621-2016	866	745	39544	53.22	85
C-4007-2013	619	619	37172	60.15	84
D-4476-2013	520	520	27898	53.65	84
C-4007-2013	619	619	37125	60.07	83
C-7679-2016	611	298	34566	115.99	83
G-9573-2012	755	755	25892	34.29	80
J-4630-2016	2718	2630	42617	16.20	80
G-5410-2012	784	310	20400	65.81	77
L-4142-2016	501	488	28821	59.30	77
B-2309-2010	1275	884	24380	27.58	76
B-5403-2013	444	444	28660	64.70	76
G-6218-2012	446	434	27145	62.98	76
D-1707-2012	284	284	22006	77.49	75
E-8874-2010	459	321	18207	56.72	75
M-6631-2013	777	710	23832	33.66	75
K-6137-2016	237	219	37949	173.28	73
F-2141-2010	264	229	25332	110.62	71
F-2141-2010	265	230	25334	110.15	71
D-2408-2015	419	419	21673	51.85	69
F-7161-2010	348	321	22952	71.50	69
F-7161-2010	348	321	22983	71.60	69
F-7161-2010	194	194	14769	76.13	69
L-2418-2014	545	541	18434	34.07	69
M-3090-2014	254	254	28564	112.46	68
K-6892-2013	753	527	24014	46.09	68
L-3893-2016	540	540	19042	35.26	67
E-6926-2013	540	540	19115	35.40	67
E-6926-2013	172	158	22547	142.70	67
G-6877-2012	828	653	14993	23.03	63
I-7972-2012	185	169	18890	111.78	62
B-6589-2008	474	454	13153	28.97	62
J-5803-2014	474	454	13181	29.03	62
J-5803-2014	372	372	20303	54.58	62
K-1918-2012	372	372	20325	54.64	62
O-7715-2015	202	202	13956	69.09	62
C-9336-2011	590	431	11811	27.86	61
F-8692-2017	504	504	16341	32.42	61
H-1692-2011	665	645	16647	25.81	61

C-3323-2013	499	499	14448	28.95	60
D-3175-2013	372	371	12043	32.46	60
I-7663-2012	256	249	12957	52.46	60
J-5104-2015	1169	1117	18297	16.38	60

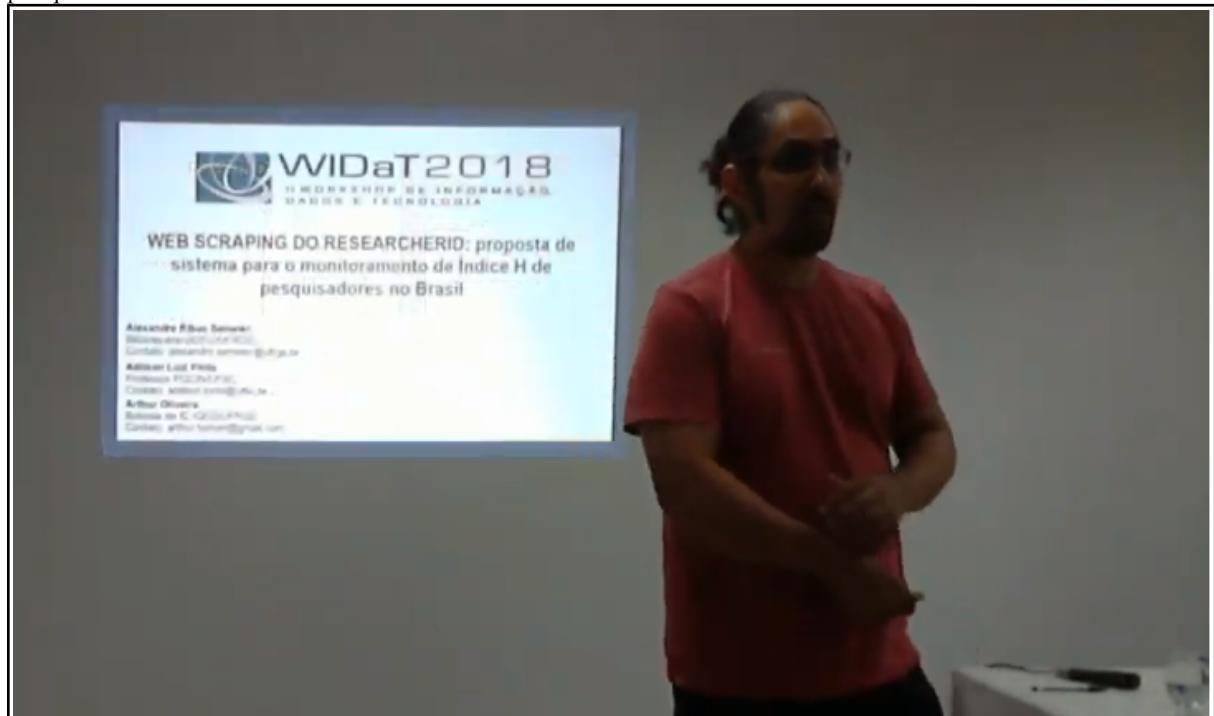
Fonte: Dados da pesquisa (2018).

Apêndice E – Validação do Índice H no Lattes CNPq

Fonte: Dados da Pesquisa (2018).

Vídeo da apresentação

Título: Web Scraping do ResearchID: proposta de sistema para o monitoramento de Índice H de pesquisadores no Brasil.



Disponível em: http://dadosabertos.info/enhanced_publications/idt/video.php?id=48

Transcrição da apresentação

Bom, primeiro eu gostaria de falar que este trabalho inicialmente, bom sou pós-doutorando do Programa de Pós-Graduação de Ciência da Informação da UFSC, sou bibliotecário do instituto de geociênciа da UFSC, e esse trabalho é uma aplicação prática de um serviço que eu estou propondo para minha universidade para que ele componha uma parte de um documento diário para avaliação de docentes que vão se candidatar para progressão de titular e para alinhar e comparar programas de pós-graduação. No entanto, eu não posso utilizar os dados porque é um trabalho que está sendo desenvolvido, eu vou falar sobre o índice nacional, o índice H.

A primeira parte a gente tem uma ferramenta desenvolvida pela Thompson que oferece a versão do índice H para todo mundo, e o Brasil é o país que mais utiliza a ferramenta para gerar o Índice H, muitas vezes para instituições, para escolas, mas também para pessoas que é o objetivo da ferramenta, são 108 mil pesquisadores brasileiros que utilizam a ferramenta.

O CNPQ adota esta ferramenta para validar o índice H ou fator H como o CNPQ chama, desses pesquisadores brasileiros, geralmente esse fator ele é mais válido para pessoas que têm mais tempo, para pesquisadores no caso, mas ela fica dentro da academia e o CNPQ usa isso para pautar a distribuição de bolsa de produtividade e para distribuição de fomento para projeto de pesquisa. Na área de geociênciа que é a área que eu tenho maior conhecimento.

Enfim, essa é a ferramenta adotada pelo CNPQ, ela fornece essas informações sobre a produção e o impacto científico que ela tem internacionalmente, é um indicador usado para medir a internacionalização da produção científica. Então qual é o objetivo do trabalho? Desenvolver uma forma automatizada para extrair esse índice H dos pesquisadores brasileiros, fazendo isso com a linguagem de programação Python e identificar esse índice H em uma frente de pesquisa nacional, quais são os pesquisadores brasileiros que possuem maior impacto em produção científica na base de dados da Thompson.

Como foi então desenvolvido esse trabalho? A pesquisa foi uma pesquisa exploratória, descritiva dos dados, de natureza quantitativa, o corpus de estudo foram 108 mil id's, identificadores, ela foi feita totalmente automatizada, foi utilizado o scraping para a raspagem dos dados, como é uma base proprietária ela não permite que a gente fique raspando ela em looping, tem que ser com calma, tempo de 10 segundos, uma navegação de humano, então para automatizar esta navegação eu utilizei algumas bibliotecas de Python, uma biblioteca para raspagem que se chama BeautifulSoup que é para fazer o scraping, e o Selenium para poder fazer essa automatização e não ficar tão agressivo e a base proibir esse acesso aos dados.

E também um multiprocessamento que é para poder, como é extraído muitos dados dela, muito tempo, a coleta sem a volumização dos dados levou em média 14 dias, é muito tempo, dividindo o processo em oito processadores da máquina, deu para reduzir o tempo da coleta para quatro dias, o que acontece essa coleta se modifica, os dados se modificam, as pessoas inserem mais informações, mais produções, e isso modifica o valor do índice, modifica a lista de publicações, então o que acontece, tem um dispositivo para falhas, então quando quebra a raspagem, quando se encerra a coleta o script retoma do último tempo válido, da última coleta válida, comparar a coleta que ele já tem e verifica se ela se modificou e reinicia-se, então ele coleta outros arquivos

diferentes, conclui a coleta e salva em um novo arquivo, a onde ele vai fazer a soma das publicações e identificar, o nome, tem nome de pessoas, primeiramente.

O código está disponível no Github, o código e a primeira extração feita em setembro, ela já se modificou de 108 mil já são 108.900 id's., Então eles abrem e coletam as métricas de informação, a quantidade de publicação, as publicações com citação e o índice H, as publicações com citação, são publicações com citação na Web of Science, quer dizer que elas não são aquelas publicações no Google, aquelas citações em outras bases de dados, somente as publicações que estão no Science Citation Index na Thompson.

Nessa coleta maior eu identifiquei a frente de pesquisa nacional, quais são os pesquisadores com pesquisa de maior impacto internacional, são pesquisadores da área de física nuclear, medicina e psicologia, esses índices são fáceis de identificar porque a frequência é um, de 108 mil pesquisadores a frequência é um, mas está distribuído, ciências humanas, enfim de um a dez, as ciências humanas ficam de um a três, de um a quatro, depende muito do pesquisador, mas a média e que a gente tem muita gente, dos 108 mil pesquisadores a maioria é zero, zero citações nessa base de dados e as pessoas não fazem bom uso às vezes da base de dados, elas usam para medir uma instituição, o script veio no caso para resolver para as pessoas que usam para medir uma instituição, você teria que colocar todas as pessoas e coletar esse grupo e verificar a média deles, então muitas pessoas fazem mal uso da plataforma, perde todo o sentido.

Então a maioria desses professores de física são titulares, são bolsistas de produtividade pelo CNPQ, para finalizar o resultado a média nacional dos 108 mil é 7,97 nosso índice H de produção.

A automatização é um processo importante, pois está em constante desenvolvimento, as pessoas estão sempre inserindo seus dados lá e o script se aplica também em pequenos grupos, no meu caso do meu trabalho que é serviço aplicado medir o programa de pós-graduação, medir o grupo de pesquisa, departamentos.

O que vai ser feito daqui para frente, desenvolver um script para validar, qual foi o problema da validação, foram as pessoas que utilizam a plataforma por brincadeira, sem compreender que o índice tem que ser validado pelo CNPQ, então eu preciso verificar quem validou sua produção no Lattes, isso já foi feito mas não deu tempo de falar aqui, uma interface gráfica e a comparação com outros países, principalmente União Soviética e Estados Unidos.

É isso, gostaria de agradecer a CAPES porque eu sou bolsista de pós-doutorado.

Muito obrigado.

Slides da apresentação

Título: Web Scraping do ResearchID: proposta de sistema para o monitoramento de Índice H de pesquisadores no Brasil.



WIDaT 2018
II WORKSHOP DE INFORMAÇÃO,
DADOS E TECNOLOGIA

WEB SCRAPING DO RESEARCHERID: proposta de sistema para o monitoramento de Índice H de pesquisadores no Brasil

Alexandre Ribas Semeler
Bibliotecário (IGEO/UFRGS)
Contato: alexandre.semeler@ufrgs.br

Adilson Luiz Pinto
Professor PGCIN/UFSC
Contato: adilson.pinto@ufsc.br

Arthur Oliveira
Bolsista de IC IGEO/UFRGS
Contato: arthur.holiver@gmail.com

Disponível em: http://dadosabertos.info/enhanced_publications/idt/presentation.php?id=48