INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Brasília
Campus Brasília

TECNOLOGIA EM SISTEMAS PARA INTERNET

Matheus Nícollas de Souza Mota
Thiago Marinho da Silva Campos

RELATÓRIO DE PRÁTICA INTEGRADA
DE
CIÊNCIA DE DADOS E INTELIGÊNCIA ARTIFICIAL

Brasília - DF
18/09/2020

Sumário

1. Objetivos

O objetivo é descobrir fatos interessantes a partir da coleta de dados de observações de OVINIs, a partir da fonte: http://www.nuforc.org/ e desenvolver um script Python que irá executar a coleta dentro de um período de vinte anos, entre setembro 1997 e agosto de 2017, armazenando tudo em um DataFrame e depois salvando em um arquivo .CSV com o nome OVNIS.csv. no GITHub.

2. Descrição do problema

O projeto consiste em reunir fatos interessantes relacionados a OVINIs, a partir de relatos realizados dentro de um período de vinte anos usando o site Nuforc. O desafio consiste em fazer a extração de dados de forma tabular, afinal, para que os dados possam ser analisados eles acabam se tornando tabelas.

O WebScraping consiste em extrair os dados formatados com tag’s da linguagem HTML. Como iremos extrair vinte anos de dados, consultaremos 240 páginas web, uma por cada mês, por vinte anos, entre setembro 1997 e agosto de 2017.

.

3. Desenvolvimento

O desenvolvimento do algoritmo foi feito na plataforma Google Collab, esta plataforma foi escolhida pois ao iniciar um notebook na mesma, as bibliotecas e dependências do Python são todas da nuvem. Os imports no código são 3, o requests é usado para pegar a url da página e ter acesso ao html da mesma, o pandas é usado para ler dataframe e planilhas, além de escreve-los também e BeaultifulSoup será usado para organizar o webscrapping identificando os elementos HTML da página. Percebemos que as URLS do site usam os anos e meses em sua formação, sendo assim criamos uma variável ano e concatenamos a indexação do ano com mês juntamente ao url da página. Existe uma estrutura de repetição 'for' que usa as variáveis vazias que serão incrementadas com todos os dados das tabelas do site, onde será gerado o csv com resultado.

3.1 Código implementado

```
# -*- coding: utf-8 -*-
"""webscraping_ovnis.ipynb
Automatically generated by Colaboratory.
Original file is located at
    https://colab.research.google.com/drive/1W-st7J7BvX0sDolO4krHV-06DIuoGl6Z
"""

import requests
import pandas as pd
from bs4 import BeautifulSoup

#inicializa variaveis de meses e anos
anos = []
meses = ['01','02','03','04','05','06','07','08','09','10','11','12']

#gerando a lista em colunas
data_hora=[]
cidade=[]
estado=[]
shape=[]
duracao=[]
descricao=[]
postado=[]
```

```
#determina o intervalo de anos a serem pesquisados (1997 - 2017)
for i in range(1997,2018):
  converte = str(i)
  anos.append(converte)

for ano in anos:
  for mes in meses:
    #Concatena URL de acordo com os meses e anos requisitados
    url = "http://www.nuforc.org/webreports/ndxe"+ano+""+mes+".html"

    #Atribui url a um request para acessar a pagina HTML e seus compon
entes
    req = requests.get(url)
    soup = BeautifulSoup(req.content, 'html.parser')

    #Coloca os conteúdos da tabela na variavel table
    table = soup.find('table')

    for row in table.findAll("tr"): #para tudo que estiver em <tr>
      dado = row.findAll('td') #variável para encontrar <td>
      if len(dado)==7: #número de colunas
        data_hora.append(dado[0].find(text=True)) #iterando sobre cada
 linha
        cidade.append(dado[1].find(text=True))
        estado.append(dado[2].find(text=True))
        shape.append(dado[3].find(text=True))
        duracao.append(dado[4].find(text=True))
        descricao.append(dado[5].find(text=True))
        postado.append(dado[6].find(text=True))

#declara dataframe
df = pd.DataFrame()

#atribui colunas as variaveis criadas
df['Date / Time']=data_hora
df['City']=cidade
df['State']=estado
df['Shape']=shape
df['Duration']=duracao
df['Summary']=descricao
df['Posted']=postado

#gera arquivo .csv
df.to_csv("ovnis.csv")

#mostra dataframe
df
```

Github: https://github.com/Prof-Fabio-Henrique/pratica-integrada-icd-e-iia-2020-1-g13-mmt

4. Considerações Finais

Ao desenvolver esta aplicação foi possível adquirir novos conhecimentos em ciência de dados, novas bibliotecas na linguagem python e ainda fazer o uso de uma ferramenta de versionamento para completo controle do avanço do projeto entre os desenvolvedores.

Referências

PANDAS. **Https://pandas.pydata.org/docs/**. Disponível em: https://pandas.pydata.org/docs/. Acesso em: 19 set. 2020.

REQUESTS. **Requests: HTTP for Humans™**. Disponível em: https://requests.readthedocs.io/en/master/. Acesso em: 19 set. 2020.

SOUP, Beautiful. **Beautiful Soup Documentation**. Disponível em: https://www.crummy.com/software/BeautifulSoup/bs4/doc/. Acesso em: 19 set. 2020.

SILVA, Réulison. **Como fazer um Web Scraping com Python**. Disponível em: https://goomore.com/blog/web-scraping-python/. Acesso em: 19 set. 2020.