INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Brasília
Campus Brasília

TECNOLOGIA EM SISTEMAS PARA INTERNET

Rosana da Silva Soares
Kennedy Viana Aguiar
Aurélio Vinícius França dos Santos

RELATÓRIO DE PRÁTICA INTEGRADA DE CIÊNCIA DE DADOS E APRENDIZADO DE MÁQUINA

Brasília - DF

6 de Março de 2021

Sumário

1. Objetivos 3

2. Descrição do problema 4

3. Desenvolvimento 5

3.1 Código implementado 5

4. Considerações finais 9

5. Referências 10

1. Objetivos

Este relatório tem como objetivo apresentar como foi realizada a extração de dados sobre o avistamento de ovnis, no período de 20 anos (1997-2017). As informações foram coletadas no site “THE NATIONAL UFO REPORTING” em forma de tabela, para isso utilizou-se a linguagem Python e bibliotecas voltadas para o manuseio de dados.

2. Descrição do problema

O site “THE NATIONAL UFO REPORTING” é um grande conjunto de dados sobre avistamento de ovnis pelo mundo afora. Nele constam o número de ocorrências de avistamento separadas por meses do ano. Dentro de cada mês os casos são mais detalhados, constando informações como: data e hora da ocorrência, formato do objeto avistado, duração do evento, relato do acontecimento e a data em que a ocorrência foi registrada no site.

O projeto de prática integrada proposto pela a equipe de docentes do IFB consiste em captar e armazenar esses dados com o intuito de, posteriormente, analisar o conteúdo captado, de modo a apresentar os dados de forma mais interativa.

3. Desenvolvimento

Os dados dispostos são visualizados por meio de uma tabela em cada página, para ter acesso a eles, utilizamos as classes **BeautifulSoup e Urlopen** das bibliotecas **bs4** e **urllib.request**, respectivamente.

Com essas bibliotecas pudemos acessar de uma forma mais fácil o conteúdo e extrair de forma simplificada os dados, estruturando-os de forma mais organizada e compreensível.

3.1 Código implementado

O primeiro passo: importar bibliotecas, a página, fazer o *web scraping* e coletar os dados.

Importando as bibliotecas e módulos necessários:

```
from bs4 import BeautifulSoup
from urllib.request import urlopen
from urllib.error import HTTPError
import csv
```

Função que capta a resposta à requisição feita ao site:

```
def getPage(url):
  try:
    resposta = urlopen(url)
  except HTTPError as e:
    return None
  return resposta
```

Formando as estruturas para armazenar os dados iniciais e criando variáveis de apoio:

```
links= []
counts=[]
counttemp = []
datas= []
dicionario = {}
```

```
link_base =" http://www.nuforc.org/webreports/ndxevent.html "
link_suporte = "http://www.nuforc.org/webreports/"
resposta_http = getPage(link_base)
objeto_soup=BeautifulSoup(resposta_http.read(),
features="html.parser")
```

Coletando da página base os links necessários para a coleta:

```
for link in objeto_soup.find_all('a')[1:]:
  if link.get_text()=='12/1996':
    break
  elif int(link.get_text()[3:])>2017:
    continue
  datas.append((link.get_text()))
  links.append((link.get('href')))

datas = datas[4:-8]
links = links[4:-8]

#Pegando os COUNTS
for link in objeto_soup.find_all('td'):
counttemp.append((link.get_text()))
counttemp = {counttemp[i]: counttemp[i+1] for i in range(0,
len(counttemp), 2)}

for i in counttemp:
  if i=='12/1996':
    break
  elif(int(i[3:])>2017):
    continue
  counts.append(counttemp[i])
counts = counts[4:-8]
```

Gerando a estrutura do ponto de partida da coleta:

```
for i in range(0, len(datas)):
  dicionario[datas[i]] = {
    "count":counts[i],
    "link":links[i]
  }

print(dicionario)
```

Percorrendo cada um dos links coletados, acessando as páginas correspondentes e extraindo todos os registros encontrados:

```
ovnis = {}
chave = 0

# percorre todo o dicionário
for i in dicionario:
  # lista que armazena as linhas das tabelas de cada página/link acessado
  linhas = []
  # acessa a página correspondente ao link em questão
  pagina = link_suporte+dicionario[i]['link']
  resposta = getPage(pagina)
  soup = BeautifulSoup(resposta.read(), features="html.parser")
  linhas_tabela = soup.find_all("tr")

  for l in linhas_tabela:
    linhas.append(l.get_text().strip().split('\n'))

  # Armazenando as informações no dicionário principal (pré-csv)
  for linha in linhas[1:]:
    ovnis[str(chave)] = {
        'data_hora': linha[0],
        'cidade': linha[1],
        'estado': linha[2],
        'formato': linha[3],
        'duracao': linha[4],
        'resumo': linha[5],
        'data_postagem': linha[6]
    }
    chave += 1
```

Convertendo a estrutura de dados gerada anteriormente em arquivo csv:

```
      csv_columns = ['ID', 'data_hora' , 'cidade', 'estado', 'formato',
'duracao', 'resumo', 'data_postagem']
      csvfile = 'OVNIS.csv'
      dict_data = [
      {'ID': i,
       'data_hora': ovnis[i]['data_hora'],
       'cidade': ovnis[i]['cidade'],
       'estado': ovnis[i]['estado'],
       'formato': ovnis[i]['formato'],
       'duracao': ovnis[i]['duracao'],
       'resumo': ovnis[i]['resumo'],
       'data_postagem': ovnis[i]['data_postagem']} for i in ovnis
      ]

      try:
        with open(csvfile, 'w') as csvfile:
          writer = csv.DictWriter(csvfile, fieldnames=csv_columns)
          writer.writeheader()
          for data in dict_data:
            writer.writerow(data)
      except IOError:
        print("I/O error")
```

Figura 1 - primeiras linhas do arquivo OVNIS.csv

OVNIS.csv

De 1 a 10 de 1367 entradas Filtro

ID	data_hora	cidade	estado	formato	duracao	resumo	data_postagem
0	8/31/17 22:00	Elizabeth	WV	Light	13 seconds	((HOAX??)) Looked like a star that moved across the sky and flashed a white light and was gone.	9/5/17
1	8/31/17 22:00	Norwalk	CT	Light	Extremely brief	Bright green light zig-zagged in the sky and disappeared after a second or two. Saw in Norwalk, CT.	9/5/17
2	8/31/17 21:00	San Diego	CA	Rectangle	30 seconds	Rectangle four white lights two red flashing lights moving slowly big large err with caution - uncertain	9/5/17
3	8/31/17 20:15	E. Rio Vista	CA	Light	20 seconds	I'm a truck driver headed E on Hwy 12 just W of I-5, when this ball of light white in color starter for my right. ((anonymous report))	9/5/17
4	8/31/17 19:30	Magna	UT	Sphere	30 minutes	Bright glowing, reflected surface. Sphere-like, seemed to change shape slightly. Hovered below clouds for 30 min.	9/5/17
5	8/31/17 10:00	Grass Valley	CA	Circle	30 seconds	I seen it twice in one night. Once with a witness bright orb traveling through the sky bright flash then disappeared. I also was taking	9/5/17
6	8/31/17 06:00	Detroit	MI	Diamond	15	Diamond shaped, silver, long. Wayne county, Michigan.	9/5/17
7	8/31/17 02:16	Lees Summit	MO	Light	1 minute	Saw two lights appearing, which looked like a plane flying towards me. I kept watching because if it was a plane, it's too big to be in	9/5/17
8	8/30/17 22:00	Henderson	NV	Circle	10 seconds	Two amber orbs observed in Henderson Nevada	9/5/17
9	8/30/17 22:00	Henderson	NV	Disk	5 minutes	Large craft seen hovering between my house in the foothills of Henderson, and Nellis AFB. It was the size of a football field, 5 lights	12/21/17

Fonte: Própria, 2021

4. Considerações Finais

Considere-se que as bibliotecas utilizadas foram úteis para o desenvolvimento do projeto, facilitando a manipulação dos dados, podendo assim ser extraído de um site, dados bagunçados transformando-os em tabelas em formato csv. As principais dificuldades encontradas foram relacionadas à implementação confusa de algumas partes da sprint, o que ocasionou certos erros e mau funcionamento do *script*, dados multiplicados e trechos desnecessários de código; no entanto, essas dificuldades foram contornadas e solucionadas, obtendo o resultado esperado ao final dos testes.

Desta forma, é relevante ressaltar que o objetivo da *sprint* foi atingido e, ainda, que houve a interação e a participação de todos os integrantes da equipe.

5. Referências

The National UFO Reporting Center. Nuforc, 2021. Disponível em: <http://www.nuforc.org/>. Acesso em: 8 de Março. 2021.

BEAUTIFUL Soup Documentation. 2020c. Disponível em: <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>. Acesso em: 8 de Mar. de 2021.