Atividade 2 Marco Carujo

July 18, 2020

[533]:
```
%load_ext lab_black
```

```
The lab_black extension is already loaded. To reload it, use:
  %reload_ext lab_black
```

0.0.1 Aluno Marco Antonio Moreira Carujo

1 ATIVIDADE 2

Utilizando os arquivos baixados no Lab 1 para indexá-los no Elasticsearch

[534]:
```
from urllib.request import urlopen
from bs4 import BeautifulSoup
from joblib import Parallel, delayed
```

[535]:
```
response = urlopen("https://sol.sbc.org.br/index.php/ctrle")  # Comprimido
response = urlopen("https://sol.sbc.org.br/index.php/ctrle/issue/view/486")  #␣
↪Completo
html = response.read()

soup = BeautifulSoup(html, "html.parser")
elements = soup.find_all("a", class_="obj_galley_link pdf")
authors = soup.find_all("div", class_="authors")
title = soup.find_all("div", class_="title")
print("Quantos artigos temos no site?", len(elements), "artigos.")
```

```
Quantos artigos temos no site? 75 artigos.
```

[536]:
```
def remove_tab_br(string):
    return string.get_text().replace("\n", "").replace("\t", "")


# Links de download (porém os arquivos já estão baixados)
links_to_view = [el["href"] for el in elements]
links_to_download = [el.replace("view", "download") for el in links_to_view]

# Nome dos arquivos
names_to_download = [el[-9:].replace("/", "-") for el in links_to_download]
```

```
# Titulos dos artigos/arquivos
title = [remove_tab_br(el) for el in title]

# Autores dos artigos/arquivos
authors = [remove_tab_br(el) for el in authors]
```

1.1 Elastic

Verificando se o ElasticSearch está online! Para isso estou utilizando um elastic search dentro da minha maquina.

[537]:
```
from elasticsearch import Elasticsearch

es = Elasticsearch()
if es.ping():
    print("ElasticSearch ONLINE !")
```

```
ElasticSearch ONLINE !
```

Vou utilizar a biblioteca python **Elasticsearch DSL** (alto nivel), cuja a documentação pode ser acessada no link aqui, para fazer indexação, mapeamento e análises dos documentos.

E tambem utilizar a biblioteca python **ElasticSearch** (baixo nivel), cuja a documentação pode ser acessada no link aqui para fazer uma busca genérica.

[538]:
```
from elasticsearch_dsl import (
    Document,
    Text,
    Date,
    Keyword,
)  # Classes para ajudar a construir o nosso documento dentro do banco
from elasticsearch_dsl import Index  # Classe de index
from elasticsearch_dsl.analysis import (
    analyzer,
    char_filter,
    tokenizer,
    token_filter,
)  # Ferramentas para construir o NLP do nosso documento
from elasticsearch_dsl.connections import connections  # Utilizando connector
```

Criando a coneção, index e mapeando uma classe

[539]:
```
connections.create_connection()  # Criando a conexão com configurações padrão
artigos = Index("artigos-index")  # Atribuindo o index 'artigos-index' a uma␣
 ↪variavel


@artigos.document
```

```
class Arquivo(Document):  # Criando a classe para ser mapeada
    titulo = Text()
    autor = Text()
    texto = Text(analyzer=pt_analyzer)


artigos.settings(number_of_shards=4)  # Configurando shards como 4

if artigos.exists():  # Caso exista o index,
    artigos.delete()  # será apagado e criado novamente
artigos.create()  # Criando o index de fato, lincando com a classe Arquivo
artigos.document(Arquivo)  # Mapeando nosso documento
artigos.get()  # Um overview da indexação
```

[539]:
```
{'artigos-index': {'aliases': {},
  'mappings': {'properties': {'autor': {'type': 'text'},
    'texto': {'type': 'text', 'analyzer': 'pt_analyzer'},
    'titulo': {'type': 'text'}}},
  'settings': {'index': {'number_of_shards': '4',
    'provided_name': 'artigos-index',
    'creation_date': '1595099450553',
    'analysis': {'filter': {'brazilian_stemmer': {'type': 'stemmer',
       'stopwords': '_brazilian_'},
      'brazilian_stop': {'type': 'stop', 'stopwords': '_brazilian_'}},
     'analyzer': {'pt_analyzer': {'filter': ['brazilian_stop',
        'brazilian_stemmer',
        'lowercase'],
       'char_filter': ['html_strip'],
       'type': 'custom',
       'tokenizer': 'standard'}}},
    'number_of_replicas': '1',
    'uuid': '7jr_KI4qQrCYK2-0TuIpFg',
    'version': {'created': '7080099'}}}}}
```

Criando um analisador de texto customizado para nosso caso de uso.

[540]:
```
char_filter_ptbr = char_filter("html_strip")  # Char para limpar resíduos de um␣
↪html
tokenizador_ptbr = tokenizer("standard")  # Tokenizador
stop_ptbr = token_filter(
    "brazilian_stop", type="stop", stopwords="_brazilian_"
)  # Analisador de stopwords
stem_ptbr = token_filter(
    "brazilian_stemmer", type="stemmer", stopwords="_brazilian_"
)  # Analisador de radical

# Nosso analisador customizado
```

```
pt_analyzer = analyzer(
    "pt_analyzer",
    tokenizer=tokenizador_ptbr,
    filter=[stop_ptbr, stem_ptbr, "lowercase"],
    char_filter=[char_filter_ptbr],
)
```

Função para pegar cada arquivo no diretório e retornar o seu conteúdo em uma string

[541]:
```
def text_by_name(name):
    arquivo = open(f"text\/{name}.txt", "r", encoding="utf-8")
    unica_string = arquivo.read().replace("\n", "")
    arquivo.close()
    return unica_string
```

Para cada arquivo salvo, crie um documento e salve no elastic search.

[542]:
```
for i, name in enumerate(names_to_download):
    Arquivo(titulo=title[i], autor=authors[i], texto=text_by_name(name)).save()
```

1.2 Buscar

Vamos procurar dentro os artigos adicionados o banco o que mais se aproxima para a query: ****“realidade aumentada”****.

[543]:
```
import pandas as pd
from IPython.display import display

# função para auxiliar o print
def eprint(response):
    results = []
    for doc in response["hits"]["hits"]:
        line = {}
        line["id"] = doc["_id"]
        line["score"] = doc["_score"]
        line["titulo"] = doc["_source"]["titulo"]
        line["autor"] = doc["_source"]["autor"]
        line["texto"] = doc["_source"]["texto"]
        results.append(line)
    display(pd.DataFrame.from_dict(results, orient="columns"))
```

Passando como busca “realidade aumentada” em qualquer campo do documento e limitando o tamanho da busca como 5 documentos

[545]:
```
import time

time.sleep(1)  # Para aguardar todos os dados serem processados e salvos.
res = es.search(q="realidade aumentada", index="artigos-index", size=5)
```

```
eprint(res)
```

```
                                          autor                    id  \
0       Eduarda Queiroz, Rafaela Moura, Ellen Souza  cTNWY3MB75dWQVrMDYpN
1  Alan Ferreira Alves, Cícero  Francisco Bezerra...  cjNWY3MB75dWQVrMDYpW
2  Deyse Mara Romualdo Soares, Gabriela Teles, Ro...  mjNWY3MB75dWQVrMDopC
3         Elvis Medeiros de Melo, Dennys Leite Maia  fjNWY3MB75dWQVrMDYqj
4  Lucas O. Lopes, Paula R. P. Oliveira, Karoline...  lzNWY3MB75dWQVrMDoow

      score                                          texto  \
0  5.175466  IV Congresso sobre Tecnologias na Educação (Ct...
1  5.140875  IV Congresso sobre Tecnologias na Educação (Ct...
2  4.053606     Tecnologias Digitais nos Processos de Ensin...
3  2.724721    O Uso de Dispositivos Móveis para o Tratamen...
4  1.325273  O "Maker" na Escola: uma Reflexão sobre Tecnol...

                                        titulo
0  Como a Realidade Aumentada tem Auxiliado no Pr...
1  Investigação de Novas Estratégias para o Ensin...
2  Tecnologias Digitais nos Processos de Ensino e...
3  O Uso de Dispositivos Móveis para o Tratamento...
4  O "Maker" na Escola: uma Reflexão sobre Tecnol...
```