



Cinthyá Lins (cml2)
Eládia Cristina (ecmac)
Gabriela Leal (glm)

Índice Invertido

Índice Invertido

- Documentos avaliados no primeiro projeto
- Tratamento dos textos das páginas que foram analisadas no primeiro
 - 102 páginas
 - lower case, stop words, plural
- É criado um DataFrame com os termos e os códigos dos documentos

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	
abaixo	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
abordagem	0.0	0.0	0.0	0.0	0.0	0.0	0.0	7.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
abrir	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
abuso	8.0	8.0	0.0	8.0	8.0	8.0	8.0	8.0	8.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
acabam	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
...	
último	0.0	0.0	0.0	0.0	1.0	0.0	2.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.0	0.0	0.0	2.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0		
única	1.0	1.0	1.0	1.0	1.0	1.0	1.0	2.0	1.0	3.0	2.0	3.0	2.0	2.0	2.0	2.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
único	0.0	0.0	0.0	0.0	0.0	1.0	9.0	0.0	0.0	0.0	0.0	3.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
úteis	1.0	1.0	0.0	1.0	1.0	1.0	1.0	2.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		
útil	16.0	12.0	0.0	13.0	13.0	12.0	16.0	12.0	16.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0		
2714 rows x 102 columns																																							

Posting

- Posting:
 - Para cada linha no dataframe, se o valor da célula for diferente de zero, a coluna era adicionada no posting.
 - Função: `to_list (dataframe)`
- Compressão com intervalo
 - Função: `compr_posting (lst)`

	Terms	Freq	Posting	Compress Posting
word				
0	abaixo	38	[0, 1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14...	[0, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
1	abordagem	3	[7, 64, 74]	[7, 57, 10]
2	abrir	20	[82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 9...	[82, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
3	abuso	8	[0, 1, 3, 4, 5, 6, 7, 8]	[0, 1, 2, 1, 1, 1, 1, 1]
4	acabam	6	[6, 49, 52, 55, 56, 57]	[6, 43, 3, 3, 1, 1]

	word	Field Text	Freq	Posting	Compress Posting
0	0	1808.title	1	[0]	[0]
1	1	1900.title	1	[4]	[4]
2	0	1967.publishing	1	[29]	[29]
3	1	1984.publishing	1	[32]	[32]
4	2	1987.publishing	1	[34]	[34]

Memory_usage

```
[133] space_ii = indice_invertido_all_view_pages.memory_usage(index=True,deep=True)
```

```
[134] space_iift = field_texts.memory_usage(index=True,deep=True)
```

```
[137] # diff all texts  
      diff_space_ii = space_ii['Posting'] - space_ii['Compress Posting']  
      diff_space_ii
```

```
↳ -4680
```

```
[138] # diff field text  
      diff_space_iift = space_iift['Posting'] - space_iift['Compress Posting']  
      diff_space_iift
```

```
↳ 816
```

Código Gamma

Função: gamma_code(num)

```
def gamma_code(num):  
    bin_n = int(bin(num)[2:])  
    offset = str(bin_n)[1:]  
    len_offset = len(offset)  
    ones_list = [1 for i in range(len_offset)]  
    unary = ''.join(map(str, ones_list))  
    length = unary + '0'  
    gamma = int(length + offset)  
  
    return (gamma)
```

Código Gamma

Memory usage ft:

Index	128
word	5712
Field Text	52452
Freq	5712
Posting	58944
Compress Posting	58128
Gamma Code	58128
dtype: int64	

Memory usage all texts:

Index	103632
Terms	185194
Freq	21712
Posting	529152
Compress Posting	533832
Gamma Code	533832
dtype: int64	

Processamento da consulta

Consulta do usuário

Consulta do usuário:

- Busca livre: “diário garoto”
- Busca estruturada:
 - { título: ‘sítio do picapau amarelo’, autor: ‘monteiro lobato’}

Leitura dos Postings

- A leitura é feita pela forma term-at-a-time.

Para cada termo da busca, a lista de documentos em que o termo está presente é colocada em uma lista com a união de todos os postings por termo na busca. Os documentos repetidos na lista são ignorados.

Ranking dos documentos

- Usando a frequência (tf) para gerar os scores
 - Calcula a frequência de cada termo por documento
- Usando o tf-idf para gerar os scores
 - Calcula a frequência de cada termo por documento e multiplica por $\log(N/n_i)$ em que N é o total de documentos na base e n_i é quantidade de documentos em que o termo aparece

Ranking dos documentos

- Cria um vetor com a frequência de cada termo da consulta para cada documento e um para a consulta
- Calcula a similaridade do cosseno entre o vetor da consulta e o vetor de cada documento

Ranking dos documentos

- Usa o valor da similaridade do cosseno para fazer o ranking
- São retornados até 10 documentos com os maiores valores de similaridade do cosseno

Correlação de consultas

- consulta 1: 'diário de um garoto'
 - tf: ['2', '9', '69', '27', '66', '92', '1', '10', '11', '12']
 - tf-idf: ['2', '9', '27', '66', '69', '92', '1', '10', '11', '12']
 - correlação de spearman: 0.96
- consulta 2: 'vidas secas'
 - tf: ['26', '10', '1', '2', '4', '9', '12', '13', '14', '19']
 - tf-idf: ['26', '10', '1', '2', '4', '9', '12', '13', '14', '19']
 - correlação de spearman: 1.0
- consulta 3: 'universo'
 - tf: ['3', '10', '14', '16', '17', '34', '78', '92', '94', '96']
 - tf-idf: ['3', '10', '14', '16', '17', '34', '78', '92', '94', '96']
 - correlação de spearman: 1.0

Correlação de consultas

- consulta 4: 'redes de computadores'
 - tf: ['7', '82', '83', '84', '85', '86', '87', '88', '89', '90']
 - tf-idf: ['7', '82', '83', '84', '85', '86', '87', '88', '89', '90']
 - correlação de spearman: 1.0
- consulta 5: 'stephen king'
 - tf: ['83', '92', '93', '94', '95', '97', '98', '99', '12', '100']
 - tf-idf: ['83', '92', '93', '94', '95', '97', '98', '99', '12', '100']
 - correlação de spearman: 1.0

Composição da Resposta

Estrutura dos dados

```
livro = {  
    author: 'string',  
    title: 'string',  
    publisher: 'string',  
    language: 'string',  
    isbn: 'string',  
}
```

```
ranking = [{  
    author: 'string',  
    title: 'string',  
    url: 'string'  
}]
```


Interface (Django)



Busca simples

Buscar

Ir para busca por atributos



Busca por características

Preencha ao menos um dos campos

Titulo:

Autor:

ISBN:

Editora:

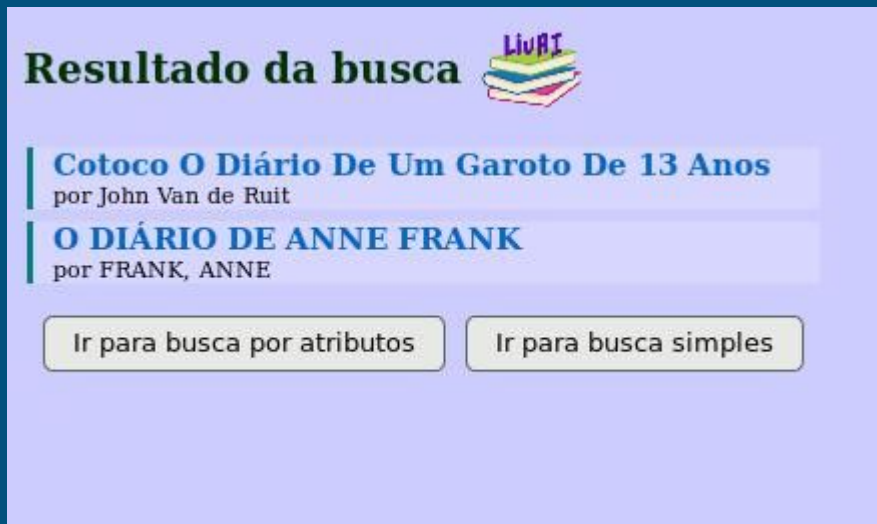
Idioma:

Buscar

Ir para busca simples

Interface (Django)

- Pesquisa por “diário”



Interface (Django)

- Buscando pelo autor “Luccas”

Resultado da busca 

Aventuras Na Netoland Com Luccas Neto
por Neto, Luccas

Luccas Neto em os aventureiros
por Luccas Neto

[Ir para busca por atributos](#) [Ir para busca simples](#)