MBA em Ciência de Dados

Técnicas Avançadas para Captura e Tratamento de Dados

Identificação e Extração de Texto

Avaliação

Material Produzido por Luis Gustavo Nonato

Cemeai - ICMC/USP São Carlos

Os exercícios a seguir farão uso do aquivo `nfe-avaliacao.pdf`, disponíveis para download no Moodle.

Exercício 1)

Considere a nota fiscal eletrônica representada no arquivo `nfe-avaliacao.pdf`. Converta o arquivo PDF em uma imagem no formato PNG. Utilize o pacote PIL para carregar a imagem gerada. Qual a resolução da imagem gerada?

a) 1653 X 2339
b) 1024 X 640
c) 640 X 2339
d) 1001 X 2020

Dica: Utilize o atribute `size` do objeto PIL para obter as dimenões da imagem e empregue os parâmetros default do método convert_from_path para gerar a imagem.

In [1]:
```
from pdf2image import convert_from_path
from PIL import Image

filename = 'nfe-avaliacao.pdf'

nfe_imagem = convert_from_path(filename)

for i,pagina in enumerate(nfe_imagem):
    # Salavando a imagen da página em um arquivo
    pagina.save('nfe-avaliacao.png', 'PNG')
```

In [2]:
```
# carregando a imagem e verificando as dimensões

nfe_image = Image.open('nfe-avaliacao.png')
print('Dimensões da imagem',nfe_image.size)
```

```
Dimensões da imagem (1653, 2339)
```

Exercício 2)

Aplique OCR para extrair o texto contido na imagem gerada no exercício 1). Quantas linhas o texto resultante possui?

a) 230
b) 231
c) 232
d) 233

Dica: Para contar as linhas, procure pelo número de ocorrências do símbolo '\n' no texto extraído.

In [3]:
```
import pytesseract as ocr
text_nfe = ocr.image_to_string(Image.open('nfe-avaliacao.png'), lang='por')

# solucao 1
nlinhas = [1 if c=='\n' else 0 for c in text_nfe]
print(sum(nlinhas)+1)

# solucao 2
print(text_nfe.count('\n')+1)

# solução 3
nlinhas = text_nfe.split('\n')
print(len(nlinhas))

# solução 4
nlinhas = text_nfe.splitlines()
print(len(nlinhas))
```

```
232
232
232
232
```

Exercício 3)

Escreva uma expressão regular para encontrar todos os valores financeiros descritos na nota. Ou seja, todos as ocorrências de uma sequência de dígitos que, precedem uma vírgula, a qual é seguida de exatamente outros dois dígitos (por exemplo: 7545,43). Quantas ocorrências de valores financeiros existem na NEF?

a) 32
b) 33
c) 36
d) 38

In [4]:
```
import re

expreg = '[\d]+,\d\d\s'

valores = re.findall(expreg,text_nfe)
print(valores)
print(len(valores))
```

```
['0,00 ', '0,00 ', '0,00 ', '0,00 ', '5687,62\n', '250,00 ', '0,00 ', '337,62 ',
'0,00 ', '0,00 ', '1425,19 ', '5600,00\n', '8,58 ', '67,90 ', '582,58 ', '0,00
', '0,00 ', '0,00 ', '0,00\n', '16,66 ', '134,90 ', '2247,43 ', '0,00 ', '0,00
', '0,00 ', '0,00\n', '51,12 ', '55,90 ', '2857,61 ', '0,00 ', '0,00 ', '0,00 ',
'0,00\n', '0,00 ', '0,00 ', '0,00\n']
36
```

Exercício 4)

Dos valores financeiros obtidos no exercício anterior, os de valor maior que R$ 100,00 somam:

a) 17456.65
b) 19122.95
c) 36456.25
d) 20345.86

Dica: Converta as strings para float.

In [6]:
```
# removendo espacos em branco, \n, \t, etc
print('Removendo espaço em branco, quebra de linha, tabulações, etc..')
float_list = [''.join(s.split()) for s in valores]
print(float_list)

# removendo '.' e substituindo ',' por '.'
print("\nRemovendo '.' e substituindo ',' por '.' ")
float_list = [s.replace('.','').replace(',','.') for s in float_list]
print(float_list)

# convertendo para float e somando
print("\nConvertendo para float")
float_list = [float(s) for s in float_list]
float_list_gt100 = [v for v in float_list if v>100.0]
print(float_list_gt100)
print('\nSoma total: ',sum(float_list_gt100))
```

```
Removendo espaço em branco, quebra de linha, tabulações, etc..
['0,00', '0,00', '0,00', '0,00', '5687,62', '250,00', '0,00', '337,62', '0,00',
'0,00', '1425,19', '5600,00', '8,58', '67,90', '582,58', '0,00', '0,00', '0,00',
'0,00', '16,66', '134,90', '2247,43', '0,00', '0,00', '0,00', '0,00', '51,12', '
55,90', '2857,61', '0,00', '0,00', '0,00', '0,00', '0,00', '0,00', '0,00']

Removendo '.' e substituindo ',' por '.'
['0.00', '0.00', '0.00', '0.00', '5687.62', '250.00', '0.00', '337.62', '0.00',
'0.00', '1425.19', '5600.00', '8.58', '67.90', '582.58', '0.00', '0.00', '0.00',
'0.00', '16.66', '134.90', '2247.43', '0.00', '0.00', '0.00', '0.00', '51.12', '
55.90', '2857.61', '0.00', '0.00', '0.00', '0.00', '0.00', '0.00', '0.00']

Convertendo para float
[5687.62, 250.0, 337.62, 1425.19, 5600.0, 582.58, 134.9, 2247.43, 2857.61]

Soma total:  19122.95
```

Exercício 5)

Encontre todas as ocorrências da palavra "VALOR" onde o OCR reconheceu o caractere "V" de forma errada. Por exemplo, existem ocorrências onde o caractere "V" foi trocado pelo símbolo "'". Quantas ocorrências foram encontrada?

a) 7
b) 8
c) 9
d) 10

Dica: utilize o símbolo '^' combinado com '[ ]'

In [7]:
```
exp_valor = r'[^V]ALOR'
V_errado = re.findall(exp_valor,text_nfe)
print(V_errado)
print(len(V_errado))
```

```
["'ALOR", "'ALOR", "'ALOR", '2ALOR', "'ALOR", 'gALOR', "'ALOR", "'ALOR"]
8
```