Scripting no Processamento de Língua Natural

Teste, 2022-06-07

1 Python

1. Dada uma lista de palavras, defina uma função utilizando listas ou dicionários em compreensão que calcula as vogais de cada palavra e imprime o resultado ordenado pelas vogais calculadas.

Considere o seguinte conjunto de palavras:

```
words = ['atribuição', 'unidirecional', 'Alunos', 'Área', 'café', 'esdrúxula']
```

O resultado esperado seria:

```
café: ae
Área: aea
atribuição: aiuiao
Alunos: auo
esdrúxula: euua
unidirecional: uiieioa
```

2. Qual é a diferença entre as seguintes linhas de código? Qual delas devolve a lista maior?

```
>>> sorted(set([w.lower() for w in text1]))
>>> sorted([w.lower() for w in set(text1)])
```

---

2 Spacy

1. Crie um programa em Python que, dado o texto de um livro como input, calcula o número de ocorrências das personagens desse livro.
2. Crie outro programa em Python que, dado o texto de um livro como input, calcula o grau de relação entre as personagens desse livro. Sugestão: como critério de relação entre personagens, pode considerar que personagens que apareçam na mesma frase estão relacionadas entre si. O resultado deve ser guardado num ficheiro JSON.

---

3 Web Scrapping, BS4

Implemente um programa em Python que dado o URL de um plano de estudos dum curso, extraia o nome e descrição de todas as Unidades curriculares desse curso.

Tome como exemplo as páginas HTML exemplo apresentadas abaixo (um plano de estudos e uma UC).

Exemplo de invocação `get-ucs https://www.uminho.pt/cursos/lei/ucs`

Exemplo de página do plano de estudos: (https://www.uminho.pt/cursos/lei/ucs)

```
<div class="uc_table_div">
  <table>
    <tr>
      <th>Regime</th>
      <th>Unidade Curricular</th>
      <th>ECTS</th>
    </tr>
    <tr>
      <td>1º Semestre</td>
      <td> <a href="https://www.uminho.pt/cursos/lei/ucs/liI"> Laboratórios de Informática I</a> </td>
      <td>5</td>
    </tr>
    <tr>
      <td>2º Semestre</td>
      <td> <a href="https://www.uminho.pt/cursos/lei/ucs/logica"> Lógica </a> </td>
      <td>5</td>
    </tr>

    (...)

  </table>
</div>
```

Exemplo de página da Unidade Curricular (https://www.uminho.pt/cursos/lei/ucs/logica)

```
<div class ="container">
    Ano letivo 2021-2022
</div>
<div class="container">
    <div class="uc_title">
        <h1> Unidade Curricular da Universidade do Minho <h1>
        <h2> <b> Lógica </b> </h2>
    </div>
    <div class="uc_descricao">
        <p>
            Esta UC está organizada apenas em práticas laboratoriais (PL) que serão usadas
            para os alunos explorarem os diferentes conceitos a introduzir em cada aula. (...)
        <p>
    </div>
</div>
```

4 Expressões Regulares, Listas em Português

Em Português as sequências são normalmente escritas como `elemento1, elemento2, ... e elementon`. Construa um função Python `getseq` que dado um texto escreva na saída as sequências de 3 ou mais elementos mono-palavra, usando "|" como separador. Exemplo: Dado o seguinte texto, t1:

```
Portugal, Espanha e França resolveram proibir a pesca durante os meses de Janeiro,
Fevereiro, Março e Abril.
Durante esses meses, a Itália e Grécia permitem pesca, arrasto, captura.
```

pretende-se que `getseq(t1)` escreva:

```
Portugal|Espanha|França
Janeiro|Fevereiro|Março|Abril
```

5 Espaços em falta

Após uma operação de OCR, um determinado texto perdeu completamente os espaços. Pretendemos construir uma ferramenta que reponha os espaços em falta. Suponha ainda que:

- Dispomos ainda de um dicionário pt (`isvalid(word)` → True or False)
- As palavras têm dimensão menor que 15 caracteres.

```
                  1                   2                   3                   4
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2
a c a v a l o d a d o , n ã o s e o l h a o d e n t e D i t a d o s F N A C 1 9 9 9 .
-   -   -   -   -   -       -   - -     - -   -     -       -   -       -     w1 (len=1)
  ----------- --- ---         ---           ---   ---         ---             w6 w2
        -----           -----       ----- -----                               w3
  -------     -------   -------   -------     ------- -------       -------   w4
    -------                                 --------- -------------           w4 w5 w7
```

1. Alguns elementos textuais permitem "adivinhar" fronteiras de palavras, permitindo dividir o texto em sub-blocos mais pequenos. Quais são os padrões que permitem essas quebras? Construa uma função que dado um texto, devolva a lista dos sub-blocos a analisar.

2. Crie uma função que construa o grafo de palavras(GP). O GP é um dicionário (início → (fim+1, pal)*) ou seja, a chave é o índice início de palavra e o valor associado é uma lista de pares (fim+1, palavra)

```
gp = mkgp("acavalodado")
{ 0 : [ (1, a) ]
  1 : [ (5, cava), (7, cavalo)]
  2 : [ (3, a), (6, aval)]
  4 : [ (5, a), (7, alo)]
  6 : [ (7, o) ]
  7 : [ (9, da), (11, dado)]
  9 : [ (11, do) ]
}
```

3. Crie uma função que dado um GP e uma posição inicial, e uma posição terminal, calcule uma frase válida, dando prioridade às palavras mais longas. `frase(gp, 0 , 11)` poderia dar "a cavalo dado"

6 Directory Walk (os.walk)

Para guardar um arquivo, optou-se por criar uma ontologia dispersa em árvore de diretorias. Cada classe X, fica pasta "C-X". Sub-classes ficam em sub-pastas. Cada individuo Y de uma class Z, fica uma pasta "I-Y" (que vai conter os ficheiros a ele associados), Esta pasta fica dentro da pasta C-Z.

Segue um exemplo de uma árvore de diretorias arquivo, e em comentário, notas acerca da ontologia associada (classes, indivíduos e triplos).

```
• A-Eurico_Tomás_Lima     # arquivo: Eurico Tomás de Lima || triplos
  • C-doc                 # class: doc       || (doc, a, class)
    • C-foto              # class: foto      || (foto, a, class) (foto, isSubclass, doc)
    • C-carta             # class: carta     || (carta, a, class) (carta, isSubclass, doc)
      • I-c1              # individuo: c1    || (c1, a, carta)
        - meta-c1.yaml    # meta de c1       || (c1, meta, "meta-c1.yaml")
        - img-c1.jpg      #                  || (c1, img, "img-c1.jpg")
      • I-c2              # ... outro individuo carta
        ....
      • C-postal          # (postal, a, class) (postal, isSubclass, carta)
    • C-partitura
```

1. Dado uma diretoria arquivo (Ex: A-Eurico_Tomás_Lima) imprima todos os indivíduos nela presentes e sua classe ou seja os pares (Individuo, class).
2. Dado uma diretoria arquivo (Ex: A-Eurico_Tomás_Lima) crie um html contendo a taxonomia das classes em listas aninhadas. Junto a cada classe mostre os seus indivíduos (entre "[]")

```
EURICO TOMÁS LIMA              ## <h1>Eurico....</h1>
Classes:
  • doc                        ## <ul><li> doc
    • foto                     ##       <ul><li> foto </li>
    • carta [c1, c2]           ##           <li> carta [c1,c2]
      • postal                 ##              <ul><li> postal </li></ul></li>
    • partitura                ##           <li> partitura</li>
```

3. Dado uma diretoria arquivo devolva o conjunto dos triplos a ela associados `triplos("A-Eurico_Tomás_Lima")` devolve `[(doc, a, class) (carta, a, class) (carta, isSubclass, doc) (c1 a carta) ...]`

Sugestão: Considere a função `os.walk` que dada uma diretoria, visita todas as suas subdiretoria

```
os.walk = walk(dirraiz, topdown=True)
    Directory tree generator.

    Procura todas as diretorias nessa árvore de subdiretoria, (incluindo a raiz)
    e devolve (yields) uma lista de triplos (dirpath, dirnames, filenames)

       dirpath is a string, the path to the directory.
       dirnames is a list of the names of the subdirectories in dirpath.
       filenames is a list of the names of the non-directory files in dirpath.

    Note that the names in the lists are just names, with no path components.
    To get a full path (which begins with top) to a file or directory in
    dirpath, do os.path.join(dirpath, name).

    Se o argumento 'topdown' for true ou não especificado, visita topdown senão: bottom-up

Uso típico:
    for dir, dirs, files in os.walk("/home/jj/spln"):
       print(f'Pasta {dir} contem: {len(dirs)} pastas e {len(files)} ficheiros')
```