

Prácticas de SAR

Sistemas de Almacenamiento y Recuperación de información

Práctica 4: NLTK

NLTK

- Para el uso de la herramienta NLTK se recomienda la lectura de <http://www.nltk.org/book>
- Como ayuda para resolver los ejercicios propuestos se ha elaborado el documento **Guia_NLTK**.
- Se debe entregar un único programa en Python que resuelva todas las acciones propuestas en los 3 ejercicios.
- Además, se deben responder por escrito al subir la tarea a la pregunta 12 del ejercicio 1 y a la pregunta 13 del ejercicio 3.

Escribe las instrucciones de Python adecuadas para realizar las acciones propuestas en cada apartado donde se adjunta el resultado correcto de su ejecución, si procede.

- 1 Acceder al corpus en castellano `cess_esp`
- 2 Mostrar el número de palabras que contiene este corpus

192685

- 3 Mostrar el número de frases que contiene

6030

- 4 Obtener las frecuencias de aparición de los ítems que componen el primer fichero del corpus anterior. Un ítem es un par (key, value) donde key es la palabra y value es la frecuencia de aparición de la palabra. Visualizar los 20 más frecuentes.

```
[('de', 23), (',', 12), ('la', 12), ('en', 9), ('y', 8), ('.', 6), ('-Fpa-', 5), ('-Fpt-', 5), ('EDF', 5), ('para', 5), ('una', 5), ('como', 4), ('con', 4), ('millones', 4), ('que', 4), ('*0*', 3), ('EAA', 3), ('a', 3), ('central', 3), ('gas', 3)]
```

6 Obtener el vocabulario del primer fichero del corpus (ordenado por frecuencia).

```
[ 'de', 'la', ',', 'en', 'y', '.', '-Fpt-', 'una', 'EDF', '-Fpa-', 'para', 'millones',
  'como', 'que', 'con', 'EAA', 'por', '*0*', 'gas', 'central', 'megavattios',
  'a', '495', 'euros', 'México', 'natural', 'potencia', 'Río_Bravo', 'Saltillo',
  'se', 'construcción', 'dólares', 'el', 'Altamira_2', 'principal', 'utilizació',
  'n', 'previsto', 'electricidad', 'Tampico', 'en_virtud_de', 'pública', 'norte',
  'portavoz', 'explotarla', 'no', 'explicó', 'duración', 'poner_en_marcha',
  'energía', 'anunció', 'funcionar', '25', 'empresa', 'estatal', 'revelar',
  'combinado', 'participaron', 'creada', 'posteriormente', 'hoy', 'CFE',
  'Electricité_de_France', 'japonés', 'primera', 'Mitsubishi', 'quedaron', 'Una',
  '1998', 'eléctricas', 'del', 'compañía', 'EFE', '51_por_ciento', 'construir',
  'participación', 'prevé', 'cuánto', 'cada', 'eléctrica', 'La', 'mayo_del_2002',
  'licitación', 'proyecto', 'invertir', 'ciclo', 'dos', '194', 'combustible',
  'acuerdo', '28', '134', 'prevista', 'quiso', 'Electricidad_Águila_de_Altamira',
  'francesa', 'funcionará', 'Comisión_Federal_de_Electricidad', 'centrales',
  'al', 'El', 'cuya', 'pagó', 'licencias', '186', 'grupo', 'japonesa', 'años',
  'empezar', 'jueves', '247', 'mayoritaria', 'red', 'un', 'accionista', 'compra',
  'Tuxpán', 'su', 'producida', 'venta', ':', 'pasará', 'tiene', 'encargará',
  'mexicana', 'debe', 'es', 'Altamira', 'intervendrá', 'asistente',
  'Mitsubishi_Corporation']
```

- 6 Obtener de forma ordenada las palabras del vocabulario de longitud mayor que 7 y que aparezcan más de 2 veces en el primer fichero del corpus.

```
['megavatios', 'millones']
```

- 7 Obtener la frecuencia de aparición de las palabras en el primer fichero del corpus. Además, y para el mismo fichero obtener la frecuencia de la palabra 'a'.

```
[23, 12, 12, 9, 8, 6, 5, 5, 5, 5, 5, 4, 4, 4, 4, 3, 3, 3, 3, 3, 3, 3, 2, 2, 2,
 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
 1, 1, 1, 1, 1, 1]
```

Freq aparición de la preposición a: 3

- 8 Obtener el número de palabras que sólo aparecen una vez en el primer fichero del corpus.

Número de palabras que aparecen una sólo vez: 95

- 9 Obtener la palabra más frecuente del primer fichero del corpus.

La palabra más frecuente es: "de"

- 10 Cargar los ficheros de PoliformaT (“spam.txt”, “quijote.txt” y “tirantloblac.txt”) como un corpus propio.
- 11 Calcular el número de palabras, el número de palabras distintas y el número de frases de los tres documentos.

```
quijote.txt 444908 24568 10822  
spam.txt 112 26 1  
tirantloblac.txt 191910 7224 8917
```

- 12 ¿Coinciden estos resultados con los de la práctica de “*Cuenta palabras*”? Justifica la respuesta.

- 1 Escribe un programa en Python para calcular cuántas veces aparecen las palabras *what*, *when*, *where*, *who* y *why* en cada una de las categorías del Corpus Brown como un diccionario donde para cada palabra tengamos la lista de categorías y la frecuencia de aparición.

```
{ 'what': [ 'adventure', 110, 'belles-lettres', 244, 'editorial', 84, 'fiction', 128,
            'government', 43, 'hobbies', 78, 'humor', 36, 'learned', 141, 'lore', 130, '
            mystery', 109, 'news', 76, 'religion', 64, 'reviews', 44, 'romance', 121, '
            science-fiction', 27],
  'when': [ 'adventure', 126, 'belles-lettres', 252, 'editorial', 103, 'fiction', 133,
            'government', 56, 'hobbies', 119, 'humor', 52, 'learned', 227, 'lore', 182, '
            mystery', 114, 'news', 128, 'religion', 53, 'reviews', 54, 'romance', 126, '
            science-fiction', 21],
  'where': [ 'adventure', 53, 'belles-lettres', 107, 'editorial', 40, 'fiction', 76, '
            government', 46, 'hobbies', 72, 'humor', 15, 'learned', 118, 'lore', 97, '
            mystery', 59, 'news', 58, 'religion', 20, 'reviews', 25, 'romance', 54, '
            science-fiction', 10],
  'who': [ 'adventure', 91, 'belles-lettres', 452, 'editorial', 172, 'fiction', 103, '
            government', 74, 'hobbies', 103, 'humor', 48, 'learned', 212, 'lore', 259, '
            mystery', 80, 'news', 268, 'religion', 100, 'reviews', 128, 'romance', 89, '
            science-fiction', 13],
  'why': [ 'adventure', 13, 'belles-lettres', 36, 'editorial', 10, 'fiction', 18, '
            government', 6, 'hobbies', 10, 'humor', 9, 'learned', 20, 'lore', 25, 'mystery
            ', 25, 'news', 9, 'religion', 14, 'reviews', 9, 'romance', 34, '
            science-fiction', 4] }
```


Escribe las instrucciones de Python adecuadas para realizar las acciones propuestas en cada apartado donde se adjunta el resultado correcto de su ejecución, si procede.

- 1 Cargar el documento “quijote.txt” en una única cadena
- 2 Mostrar todos los símbolos del documento ordenados por orden alfabético.

```
! " ' ( ) , - . 0 1 2 3 4 5 6 7 : ; ? A B C D E F G H I J L M N O P Q R S T U  
V W X Y Z ] a b c d e f g h i j l m n o p q r s t u v x y z i « » ¿ Á É Í  
Ñ Ó Ú à á é í ï ñ ó ù ú ü
```

- 3 Eliminar del texto los símbolos siguientes:

```
i! " ' ( ) , - . : ; ¿? ] « »
```

- 4 Mostrar todos los símbolos del documento filtrado ordenados por orden alfabético

```
0 1 2 3 4 5 6 7 A B C D E F G H I J L M N O P Q R S T U V W X Y Z a b c d e f g h i  
j l m n o p q r s t u v x y z Á É Í Ñ Ó Ú à á é í ï ñ ó ù ú ü
```

- 5 Obtener el número de palabras y el número de palabras distintas del texto filtrado. Mostrar la 10 primeras y las 10 últimas en orden alfabético

```
381212 24480  
10 16 1604 1614 1615 17 23 A ABC ACADÉMICO  
última últimamente últimas último últimos única único únicos útil útiles
```

- 6 Obtener las frecuencias de aparición de los ítems que componen el documento filtrado. Un ítem es un par (key, value) donde key es la palabra y value es la frecuencia de aparición de la palabra. Visualizar los primeros 20 ítems.

```
[('que', 20549), ('de', 17997), ('y', 17166), ('la', 10202), ('a', 9532), ('el', 7962), ('en', 7907), ('no', 5787), ('se', 4690), ('los', 4681), ('con', 4053), ('por', 3779), ('las', 3423), ('le', 3396), ('lo', 3393), ('su', 3320), ('don', 2538), ('del', 2465), ('me', 2345), ('como', 2244)]
```

- 7 Crear un nuevo documento eliminando las stopwords del texto filtrado.
- 8 Obtener el número de palabras y el número de palabras distintas del texto sin stopwords. Mostrar la 10 primeras y las 10 últimas en orden alfabético

```
183251 24066
10 16 1604 1614 1615 17 23 ABC ACADÉMICO ACADÉMICOS
última últimamente últimas último últimos única único únicos útil útiles
```

- 9 Obtener las frecuencias de aparición de los ítems que componen el documento sin stopwords. Visualizar los primeros 20 ítems.

```
[('don', 2538), ('Quijote', 2164), ('Sancho', 2145), ('si', 1798), ('dijo', 1789), ('tan', 1219), ('ser', 1056), ('respondió', 1053), ('bien', 964), ('señor', 948), ('así', 905), ('merced', 900), ('sino', 694), ('dos', 672), ('pues', 639), ('decir', 577), ('caballero', 573), ('hacer', 535), ('aunque', 525), ('Dios', 518)]
```

- 10 Crear un nuevo documento sustituyendo cada palabra del texto sin stopwords por su raíz. Para ello se utilizará el stemmer snowball.
- 11 Obtener el número de palabras y el número de palabras distintas del nuevo documento. Mostrar la 10 primeras y las 10 últimas en orden alfabético

```
183251 10134
```

```
10 16 1604 1614 1615 17 23 abad abadej abades
```

```
zoroastr zorr zorrund zuec zulem zumb zurd zurron zuz ñud
```

- 12 Obtener las frecuencias de aparición de los ítems que componen el nuevo documento. Visualizar los primeros 20 ítems.

```
[('don', 2656), ('quijot', 2180), ('sanch', 2158), ('si', 1966), ('dij', 1882),  
, ('señor', 1812), ('respond', 1277), ('tan', 1243), ('hac', 1158), ('buen',  
, 1115), ('asi', 1095), ('bien', 1069), ('ser', 1057), ('dec', 967), ('  
caballer', 955), ('merc', 900), ('pues', 865), ('parec', 833), ('algun',  
811), ('cos', 805)]
```

- 13 Justifica los resultados obtenidos en los pasos 5, 8 y 11.