Tarea 7

NLP

Descripción: para un texto en español, tokenización, lematización con Stanza y determinar la frecuencia de los tokens y de los lemas.

Alumno: Miguel Angel Soto Hernandez

```
!pip install stanza
```

```
#librerias necesarias
%matplotlib inline
import stanza
import numpy as np
import matplotlib.pyplot as plt

# Libreria en espanol de Stanza
stanza.download('es')
```

```
# abrimos nuestro texto y lo guardamos en la variable corpus
corpus = open('./Marques de Sade -Justine.txt').read()

print(corpus)
print(len(corpus))
```

```
    Justine, o los infortunios de la virtud

    de

    Donatien Alphonse François, Marqués de Sade


    Prólogo

    A mi buena amiga

    Sí, Constance, a ti dirijo esta obra; a la vez el ejemplo y el honor de tu sexo, sumando al

    La intención de esta novela (no tan novela como parece) es nueva sin duda; el ascendiente de

    Pero ofrecer por doquier el Vicio triunfante y la Virtud víctima de sus sacrificios; mostrar

    ¿Lo habré conseguido, Constance? ¿Provocará una lágrima de tus ojos mi triunfo? En una palab

    EXPLICACION DE LA ESTAMPA

    La Virtud, entre la Lujuria y la Irreligión. A su izquierda está la Lujuria, bajo la figura

    ¡Quién sabe, cuando el Cielo nos hiere con sus golpes, si la mayor desgracia no es un bien p

    Edipo en casa de Admeto

    ¡Oh amigo mío! La prosperidad del Crimen es como el rayo, cuyos resplandores engañosos sólo


    Primera parte
```

```
La obra maestra de la filosofía sería desarrollar los medios de que se sirve la Providencia

Si, llenos de respeto por nuestras convenciones sociales, y sin apartarnos jamás de los diqu

Tales son los sentimientos que dirigirán nuestros trabajos, y en consideración a esos motivo

Esta mujer había recibido, no obstante, la mejor educación: hija de un importantísimo banque

En esta época, fatal para la virtud de las dos jóvenes, todo lo perdieron en un solo día: un

La señora de Lorsange, entonces llamada Juliette, y de un carácter e inteligencia prácticame

Les dieron a ambas veinticuatro horas para abandonar el convento, dejándoles la tarea de ins

Utilizando otros recursos, Juliette dijo entonces a su hermana que, con la edad y la cara qu

Justine sintió horror de tales discursos; dijo que prefería la muerte a la ignominia y, pese

Por consiguiente, las dos jóvenes se separaron, sin ninguna promesa de volver a verse, dado

Mimada desde su infancia por la costurera de su madre, Justine cree que esta mujer será sens

— ¡Oh, cielos! —Dice la pobre criatura—, si es preciso que los primeros pasos que doy por el

Justine, llorosa, visita a su sacerdote; le describe su estado con el enérgico candor de su
```

```
# definiendo el idioma y las funciones que se aplicarán
stanza_nlp = stanza.Pipeline('es', processors='tokenize, lemma, pos')

# convirtiendo el texto a un formato reconocido por stanza
texto_stanza = stanza_nlp(corpus)
```

```
2021-03-28 06:32:02 WARNING: Language es package default expects mwt, which has been added
2021-03-28 06:32:02 INFO: Loading these models for language: es (Spanish):
=======================
| Processor | Package |
-----------------------
| tokenize  | ancora  |
| mwt       | ancora  |
| pos       | ancora  |
| lemma     | ancora  |
=======================

2021-03-28 06:32:02 INFO: Use device: cpu
2021-03-28 06:32:02 INFO: Loading: tokenize
2021-03-28 06:32:02 INFO: Loading: mwt
2021-03-28 06:32:02 INFO: Loading: pos
2021-03-28 06:32:03 INFO: Loading: lemma
2021-03-28 06:32:03 INFO: Done loading processors!
```

```
# por medio de ciclos for anidados obtenemos los tokens y los lemas
# de nuestro texto
tokens = np.array([palabra.text for oracion in texto_stanza.sentences
                                for palabra in oracion.words])
lemas = np.array([palabra.lemma for oracion in texto_stanza.sentences
                                for palabra in oracion.words])
print(len(tokens))
print(len(lemas))
```

```
128453
128453
```

```
# importamos re para manejar expresiones regulares
import re

#  \w+ acepta cualquier caracter que se encuentre en el rango de [a-zA-Z0-9_]
# aplicamos una expresión regular que haga
tokens = np.array([token for token in tokens if re.search('\w+', token)])
lemas = np.array([lema for lema in lemas if re.search('\w+', lema)])
```

```
# transformar palabras a minúsculas para que hagan match con las palabras auxiliares
tokens = np.array([token.lower() for token in tokens])
lemas = np.array([lemas.lower() for lemas in lemas])
```

```
# importamos spacy para obtener las palabras auxiliares del espanol
import spacy
from spacy.lang.es.stop_words import STOP_WORDS
stopwords = spacy.lang.es.stop_words.STOP_WORDS

# quitamos las palabras auxiliares mediante un bucle que las busque en el texto
tokens = np.array([token for token in tokens if not token in stopwords])
lemas = np.array([lema for lema in lemas if not lema in stopwords])
```

```
# total de tokens y lemas
print(f'Tokens: {len(tokens)}')
print(f'Lemas: {len(lemas)}')
```

```
Tokens: 47829
Lemas: 46063
```

```
# importamos nltk.probability para obtener la frecuencia
# de nuestras palabras
from nltk.probability import FreqDist

frecuencia_tokens = FreqDist(tokens)
frecuencia_lemas = FreqDist(lemas)
```

```
# Frecuencia tokens
frecuencia_tokens.most_common(100)
```

```
[('a', 2833),
 ('y', 2483),
 ('o', 301),
 ('thérèse', 209),
 ('hombre', 198),
 ('señor', 161),
 ('mujer', 149),
 ('jamás', 146),
 ('señora', 146),
 ('años', 132),
 ('naturaleza', 130),
 ('instante', 128),
 ('joven', 126),
 ('virtud', 122),
 ('casa', 116),
 ('vida', 115),
```

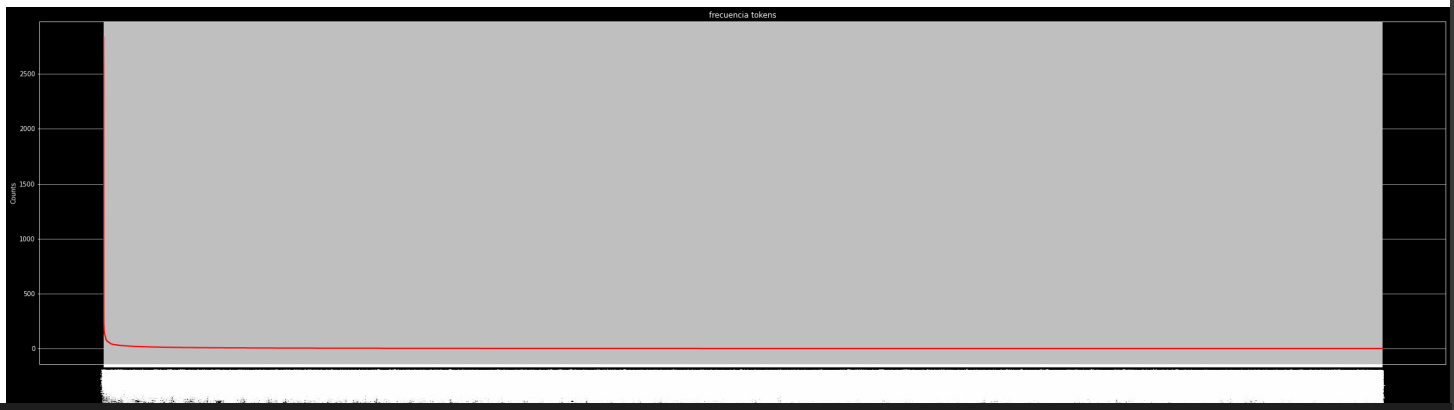
```
 ('crimen', 108),
 ('oh', 106),
 ('dios', 101),
 ('ojos', 91),
 ('conde', 87),
 ('inmediatamente', 87),
 ('mundo', 85),
 ('mujeres', 85),
 ('placeres', 84),
 ('—dijo', 83),
 ('hombres', 79),
 ('manos', 78),
 ('noche', 75),
 ('rodin', 75),
 ('duda', 74),
 ('corazón', 74),
 ('muerte', 68),
 ('mil', 67),
 ('cruel', 66),
 ('—me', 66),
 ('lágrimas', 65),
 ('fuerza', 64),
 ('boca', 64),
 ('placer', 64),
 ('hubiera', 63),
 ('leyes', 63),
 ('punto', 63),
 ('dubois', 62),
 ('criatura', 61),
 ('podía', 61),
 ('querida', 60),
 ('suerte', 60),
 ('desdichada', 59),
 ('crímenes', 59),
 ('habría', 58),
 ('cielo', 56),
 ('padre', 56),
 ('roland', 56),
 ('mano', 55),
 ('hija', 53),
 ('imposible', 53),
 ('palabra', 52),
 ('camino', 52),
```

```
# Frecuencia lemas
frecuencia_lemas.most_common(100)
```

```
[('a', 2833),
 ('y', 2483),
 ('o', 301),
 ('querer', 280),
 ('hombre', 277),
 ('mujer', 235),
 ('deber', 229),
 ('thérèse', 209),
 ('señor', 170),
 ('crimen', 167),
 ('joven', 157),
 ('servir', 157),
 ('dejar', 152),
 ('placer', 149),
 ('año', 147),
 ('virtud', 146),
 ('jamás', 146),
 ('señora', 146),
```

```
('instante', 145),
('encontrar', 143),
('acabar', 142),
('llegar', 134),
('mano', 133),
('naturaleza', 130),
('creer', 128),
('sentir', 124),
('parecer', 121),
('casa', 120),
('seguir', 119),
('vida', 115),
('ofrecer', 112),
('oh', 106),
('dios', 106),
('cuyo', 104),
('fuerza', 96),
('ojo', 92),
('espantoso', 89),
('pasar', 89),
('palabra', 88),
('cruel', 88),
('conde', 88),
('inmediatamente', 87),
('mundo', 85),
('recibir', 85),
('desdichado', 84),
('noche', 81),
('tratar', 79),
('corazón', 78),
('duda', 77),
('cosa', 76),
('abandonar', 75),
('colocar', 75),
('rodin', 75),
('volver', 74),
('ley', 74),
('descubrir', 74),
('pie', 72),
('entregar', 72),
('salir', 72),
```

```
# graficando tokens
plt.style.use('dark_background') # usar fondo negro para nuestro plot
plt.figure(figsize=(40, 10)) # definir el tamaño de nuestra figura que se mostrará
plt.title('frecuencia tokens') # título y tamaño de la letra de nuestro plot
frecuencia_tokens.plot(color='red') # plot de nuestra función
```




```
# graficando tokens
plt.style.use('dark_background') # usar fondo negro para nuestro plot
plt.figure(figsize=(40, 10)) # definir el tamaño de nuestra figura que se mostrará
plt.title('frecuencia lemas') # título y tamaño de la letra de nuestro plot
frecuencia_lemas.plot(color='red') # plot de nuestra función
```