Ciencias de la Computación | CIMAT
Procesamiento de Lenguaje Natural | Fecha de entrega:
Dr. Adrián Pastor López Monroy **T1: Python** | **4-Feb-2022 23:59:59**

Realiza los siguientes puntos en un notebook de Python lo mejor organizado y claro posible. Ponga su nombre completo al archivo de entrega (e.g., adrian_pastor_lopez_monroy.ipynb) y también en la primera celda del notebook junto con el número de Tarea. Para referencia puede apoyarse y usar el código del libro: https://www.nltk.org/book/ y algún tutorial de Python.

Al entregar la tarea, sube al classroom el notebook como un archivo (e.g., NO un zip, NO una liga a tu Drive). El notebook deberá haber sido ejecutado en tú máquina y mostrar el resultado en las celdas. Los datos de las conferencias ponlos en una carpeta en tu GDrive y SOLO comparte la liga (NO SUBIR al Classroom el zip).

1 Construcción de un corpus (0pts)

Utilice Python para construir el corpus del *Lecture 3: Python y Talacha*. El objetivo es llegar al mismo corpus que se muestra en clase de conferencias seudo limpias en texto. Siéntase libre de usar el mismo código de clase y NLTK, o bien su propio código y cualquier otra librería de Python.

Nota: Exclusivamente en este punto, SOLO proporcionar la liga a su carpeta GDrive con estos datos. No escribir en el Notebook las instrucciones para hacerlo, solo poner en una celda la liga a su GDrive, además de ponerla en el Classroom al subir la tarea.

2 Vistazo a los datos (40pts)

Utilice funciones básicas de Python para hacer lo siguiente:

1. Cargue todas las conferencias en un *string* y aplique la función *split* para generar tokens fácilmente.
2. Contar la cantidad de palabras en todas las conferencias.
3. Extraer el vocabulario y mostrar su longitud.
4. Mida la riqueza del vocabulario de alguna forma en todos los documentos.
5. Haga lo mismo que los 4 puntos anteriores pero con todo el texto en minúsculas. Vea las diferencias y comente brevemente.
6. Haga lo mismo que los puntos 1, 2 y 3 usando el tokenizador *RegExp* de NLTK, con una expresión regular que trate de sacar solo tokens que pueden contener solo letras mayúsculas y minúsculas. Después cargue los tokens en un objeto *Text* de NLTK en lugar de una lista de Python.

7. Haga todo el texto minúsculas. Haga los mismo que los puntos 1, 2, y 3 usando el tokenizador *TweetTokenizer*. Haga el resto de esta tarea asumiendo haber hecho este punto.

3 Funciones NLTK (40pts)

1. Use la función *concordance* para ver el contexto de 5 palabras que usted elija.

2. Elija una palabra que le parezca interesante y muestre palabras con uso similar. Muestre también los contextos comunes.

3. Haciendo uso de alguna librería *time* de Python: organice todos los archivos de las conferencias desde la más antigua hasta la más nueva (echando mano del nombre del archivo y fechas), y cárguelos en un objeto *Text* para generar un *dispersion plot* de las siguientes palabras: *prian, corrupción, mafia, narco, gasolina, pipas, conacyt, ciencia, turismo, pandemia, coronavirus, covid, delta, omicron, vacuna, vacunacion.* y otras tres palabras de su elección.

4. Muestre 50 colocaciones de todo el corpus.

5. Muestre un histograma de longitud en caracteres de las palabras. Muestre en el histograma el top 5 de longitudes más largas.

6. Muestre 50 palabras con longitud mayor a 8 caracteres y frecuencia mayor a 5 en todo el texto usando comprenhension list de python.

7. Ver https://www.nltk.org/book/ch02.html en la sección 1.5. Proponga una gráfica usando *ConditionalFreqDist* como la de la Sección 1.5 para estos datos. Usted elige las palabras.

8. Use la lista de stopwords de nltk y obtenga la cantidad de palabras en los datos con y sin stopword.

9. Muestre las 300 palabras más frecuentes en las conferencias, sin tomar en cuenta stopwords. Muéstrelas de la más frecuente a la menos frecuente.

4 Otras librerías en Python (20pts)

Investigue y comente brevemente en sus propias palabras:

1. Mencione dos librerías en Python además de NLTK para NLP. Ponga una desventaja y ventaja de cada una.

2. Mencione tres alternativas para Text Processing en NLP qué existen en otros lenguajes. De una ventaja y desventaja de cada una.