Informe NLP: Trabajo Práctico 1 - Parte 2. Tiny Towns.

Alumno: Adriano Marzol.

Link a Github: https://github.com/adrianomarzol/TP1Parte2\_NLP

Acondicionamiento antes de arrancar los ejercicios propuestos:

Antes de comenzar a resolver agregué el código a el archivo TP1\_TinyTowns.ipnb el código de un web scraping de la página BGG para obtener reseñas de los usuarios sobre el juego que se guardó en un archivo csv llamado reseñas\_tiny\_towns.csv.

Luego agrego un archivo llamado consultas\_tiny\_towns.xlsx en el que contienen las consultas con sus respectivas etiquetas para resolver el ejercicio 6.

Ejercicio 2:

Utilizo el texto de la sección información review\_externa.txt. Se obtienen oraciones del texto extenso utilizando la librería spacy. Luego se vectoriza cada oracion utilizando el modelo de embeddings de Hugging, distiluse-base-multilingual-cased-v1 que es bueno para tareas de búsqueda semántica.

Luego se vectoriza una serie de frases nuevas y se calculan las distancias de coseno, jaccard y dice para encontrar la oración con la menor distancia a cada nueva frase para cada tipo de medida de distancia.

Finalmente se grafica con tsne en 3 dimensiones los embeddings de las oraciones y diferenciados los embeddings de las nuevas frases ingresadas.

Ejercicio 3:

En este ejercicio utilizo el texto manual.txt también de la sección de información. De forma similar obtengo oraciones del texto extenso con spacy. Luego con el mismo spacy obtengo para cada oración los sustantivos que presenta(POS) y además categorizar entre los diferentes sustantivos obtenidos(NER). Se guarda todo en un DataFrame que contiene el sustantivo, la oración que lo contenia y la etiqueta NER.

Finalmente se busca los sustantivos que más se acercan a un sustantivo nuevo utilizando otra vez las distancias de coseno jaccard y dice.

Ejercicio 4:

Para este ejercicio uso los diferentes textos de videos también de la sección de información. Se utiliza la librería langdetect para detectar los idiomas de los textos de los diferentes videos.

Ejercicio 5:

Para este ejercicio se utilizan las reseñas de los usuarios que se mencionaban al principio del informe extraídas de la página BGG.

Se realiza un análisis de sentimiento de las diferentes reseñas utilizando las librería SentimentAnalysis(<https://pypi.org/project/SentimentAnalysis/>) que nos devuelve entre otras cosas una puntuación entre -1 y 1 y un sentimiento(Positive, Negative, Neutral). Luego se crea una función para buscar la reseña más similar a una nueva reseña filtrandose por el tipo de sentimiento de la nueva reseña.

Ejercicio 6:

Se cargan las consultas sobre el juego mencionadas al inicio del informe y se crea un set de datos en el que a cada etiqueta se la codifica con un número y a cada consulta se la vectoriza utilizando el modelo de Hugging face all-mpnet-base-v2, para luego este set de datos ser utilizado en un modelo de clasificación.

Luego estos datos son usados en un modelo de regresión logistica para tratar de predecir a que etiqueta pertenece cada consulta. Se balancean cambiando class\_weight debido a que los datos poseen estan desbalanceados hacia la etiqueta de informacion.

Luego tambien hay otra variante en la cual se utiliza tambien un modelo de regresion logistica pero en este caso la vectorizacion de las consultas no se hace a traves de un modelo de embeddings sino utilizando el vectorizador TfidfVectorizer.