Facultad de Ciencias Exactas, Ingeniería y Agrimensura

Tecnicatura Universitaria en Inteligencia Artificial

Procesamiento del Lenguaje Natural

Trabajo Práctico N°: 1

06/11/2024

Docentes:
- Geary, Alan.
- Manson, Juan Pablo.

Integrantes:
- Pace, Bruno. Legajo: P-5295/7.
- Sancho Almenar, Mariano. Legajo: S-5778/9.

Introducción

En el siguiente informe se detalla la implementación de técnicas de procesamiento del lenguaje natural para el desarrollo de un programa capaz de realizar recomendaciones a un usuario de acuerdo a su estado de ánimo y a la preferencia del mismo.
Para la realización del programa, se utilizará dos archivos .ipynb dedicados para la obtención del dataset y otro con la implementación del programa.
Para finalizar se presentará un análisis sobre los resultados obtenidos.

Metodología

Obtención del dataset

La obtención del dataset resultante será llevada a cabo por el notebook '***NLP_TP1_datasets.ipynb***' donde se aplican técnicas de procesado de archivos *.csv* y de web scrapping.
El programa trabajará con un dataset que cuenta con información sobre los libros, juegos de mesa y las películas más populares. Para la obtención del mismo, se parte desde tres datasets distintos que requieren de un procesamiento.
El resultado final del dataset requerido indica el título, una breve descripción y las categorías que abarca cada libro, película o juego de mesa.

Películas

En el dataset se encuentra un total de 1000 películas con once columnas descriptivas, con información que abarca desde título, descripción y género hasta año de lanzamiento, ganancias obtenidas o puntuación en Metacritic.

- **Rank**: Ranking de la película según su puntuación.
- **Title**: Título de la película.
- **Genre**: Género(s) de la película.
- **Description**: Breve sinopsis o resumen de la película.
- **Director**: Nombre del director de la película.
- **Actors**: Principales actores de la película.
- **Year**: Año de estreno de la película.
- **Runtime**: Duración de la película en minutos.
- **Rating**: Calificación general de la película.
- **Votes**: Número de votos recibidos por la película.
- **Revenue (Millions)**: Ingresos de la película en millones de dólares.

- **Metascore**: Puntuación promedio obtenida en Metacritic.

Para la realización del programa sólo nos interesan las columnas *Title, Description* y *Genre.*

Title	Description	Category
Coraline	An adventurous girl finds another world that i...	[Animation, Family, Fantasy]
The 9th Life of Louis Drax	A psychologist who begins working with a young...	[Mistery, Thriller]

Juegos de mesa

Al igual que el dataset de películas, aquí se encuentra con información sobre 1000 juegos de mesas. En él se describe en 17 columnas cada juego con información tal que su título, descripción, número de jugadores, etc.

- **Rank**: Ranking del juego según su puntuación.
- **Game Name**: Nombre del juego de mesa.
- **Game Href**: Enlace a la página del juego en BoardGameGeek (si aplica).
- **Geek Rating**: Puntuación del juego en BoardGameGeek.
- **Avg Rating**: Calificación promedio del juego basada en las valoraciones de los usuarios.
- **Num Voters**: Número de personas que han votado por el juego.
- **Description**: Breve descripción o sinopsis del juego.
- **Year Published**: Año en que se publicó el juego.
- **Min Players**: Número mínimo de jugadores necesarios para jugar.
- **Max Players**: Número máximo de jugadores permitidos.
- **Min Playtime**: Tiempo mínimo recomendado para jugar el juego (en minutos).
- **Max Playtime**: Tiempo máximo recomendado para jugar el juego (en minutos).
- **Min Age**: Edad mínima recomendada para jugar.
- **Avg Weight**: Promedio de la complejidad o "peso" del juego según los usuarios.
- **Best Num Players**: Número ideal de jugadores recomendado para el mejor disfrute del juego.
- **Designers**: Diseñadores del juego de mesa.
- **Mechanics**: Mecanismos o reglas principales del juego.
- **Categories**: Categorías o tipos de juego (por ejemplo, estrategia, cartas, familiar, etc.).

Sólo es de nuestro interés las columnas *Game Name, Description* y *Categories.*

Title	Description	Category

Libros

Para la obtención del dataset, se realizaron técnicas de web scrapping sobre la web '*Project Gutenberg*'.

El web scraping consiste en buscar sobre cada título presente en la página su link de referencia.
Dicho link de referencia nos indica la página del libro donde se encuentra información adicional sobre el libro como una descripción del mismo y las categorías que abarca. Luego de aplicar técnicas de web scraping sobre cada link de referencia, obtenemos un dataset que cuenta con título, descripción y categoría de los 1000 libros más descargados en el día.

Title	Description	Category

Dataset final

El dataset que será guardado y utilizado por el programa será el conjunto de los tres datasets obtenidos anteriormente, indicando además el tipo de elemento que describe (*juego/libro/película*).

Title	Description	Category	Type

Creación y funcionamiento del programa

Análisis de sentimiento

El programa se desarrolla en el notebook *'NLP_TP1_model.ipynb'*.
Para comenzar, el programa le indica al usuario que ingrese una frase para determinar su estado de ánimo. El mismo se clasificará mediante la función *get_sentiment().*
La función se encarga de recibir el texto ingresado por el usuario y mediante el uso de BERT, la clasificará como 'Alegre', 'Ni fu, ni fa' o 'Melancólico' dependiendo de la puntuación recibida por el modelo.

La función *get_sentiment()* además, se encarga de determinar el estado de ánimo que más se relaciona con el contenido a recomendar, agregando su resultado en una nueva columna 'Sentiment' en el dataset a trabajar.

Encontrar coincidencias y devolver recomendación

Una vez determinado el sentimiento, se le pide al usuario que ingrese un *prompt* donde indique con una breve descripción el contenido que le interesaría consumir.
Se compara la descripción provista con las descripciones presentes en el dataset y se evalúa las similitudes mediante el uso de *SentenceTransformer* usando el modelo preentrenado *'msmarco-MiniLM-L-6-v3'*. Este modelo fue optimizado para calcular embeddings de oraciones o frases, lo cual es útil en tareas de búsqueda semántica, recomendación de contenido y recuperación de información.

Resultados

El resultado nos devuelve la recomendación del contenido con mayor similitud, indicando su titulo, descripcion, tipo, categoria.