
Facultad de
Ciencias Exactas,
Ingeniería y Agrimensura


Universidad Nacional de Rosario

Facultad de Ciencias Exactas, Ingeniería y Agrimensura

# TRABAJO PRACTICO Nº 01

## Procesamiento del lenguaje natural - TUIA

## 06/11/2024

**Profesores**: Juan Pablo Manson, Alan Geary.

**Alumnos:**

Bravi Eugenio B-6600/1

Nemeth Ulises N-1249/1

**Indice**



# **1.** Resumen

Este trabajo presenta un sistema de recomendaciones recreativas para interiores, basado en procesamiento de lenguaje natural, que sugiere películas, libros y juegos de mesa. Utiliza un modelo de clasificación de emociones y técnicas de NLP para ofrecer opciones personalizadas.

## **2.** Introduccion

Este trabajo se centra en desarrollar un sistema de recomendaciones recreativas puertas adentro basado en procesamiento de lenguaje natural que responda a estados de ánimo específicos del usuario. El enfoque es especialmente relevante en situaciones donde las actividades al aire libre pueden estar limitadas, como unas vacaciones en la playa con días de mal tiempo, donde contar con alternativas de entretenimiento es esencial.

El objetivo de este proyecto es construir un clasificador de estados de ánimo que, a partir de una entrada en lenguaje natural, identifique el estado de ánimo del usuario y sugiera actividades recreativas acordes a sus preferencias. Las recomendaciones incluirán una selección entre distintas formas de entretenimiento, brindando opciones relevantes y personalizadas en función de lo que pida el usuario usando inputs de texto. Para lograr esto, el sistema utilizará técnicas de NLP para clasificar estados de ánimo y comparar similitudes semánticas, empleando datasets de películas, juegos de mesa y libros.

La estructura del informe está organizada de la siguiente manera: la sección **Fuentes de datos**, informando brevemente en qué consisten y de donde se obtuvieron, mientras que **Desarrollo** detalla los pasos y algoritmos necesarios para construir el clasificador y generar recomendaciones. Posteriormente, se presentan los **resultados** obtenidos, seguidos de un análisis crítico de estos. Finalmente, las **conclusiones** donde se proponen posibles mejoras y aplicaciones futuras del sistema.

## **3.** Fuentes de datos

- **bgg database.csv:** Dataset de juegos de mesa.
- **IMDB-Movie-Data.csv:** Dataset de películas.
- **Proyecto Gutenberg:** Dataset extraído a través de web scraping, contiene los 1000 libros más populares del mes.
- **sentimientos.csv:** Dataset generado manualmente para el entrenamiento de un modelo de clasificación.

# **4.** Desarrollo

## 1) Scraping

En esta etapa se realizó el scraping en la página web del **Proyecto Gutenberg** donde se extrajo los 1000 libros más populares del mes. El scraping se realizó con las librerías **BeautifulSoup** para el parseo del código html y la librería **request** para hacer solicitudes a las url.

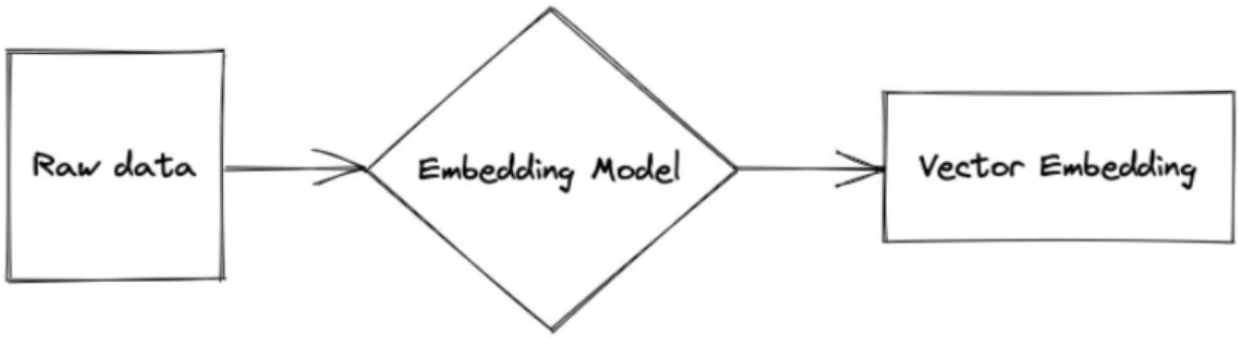
En el proceso de scraping lo primero que se hizo fue hacer una solicitud a la **url** del Proyecto Gutenberg. Luego se parseo el html con la librería **BeautifulSoup.** En el html se buscó todas las listas (“ol”) y se eligió la lista que tiene los 1000 libros más populares del mes. Una vez seleccionada la lista se recolectó todas las url a las páginas de los libros de sus elementos. Luego se realizó nuevamente un proceso de solicitud y parseo a cada uno de los links en los que se busco la tablas (“table”) con la clase “bibrec” ya que en esta se encontraban los datos de los libros. Finalmente se extrajo la información de las tablas y se guardaron en un dataframe.

## 2) Preprocesamiento de datos

En esta etapa se realizó un proceso de limpieza en los datos en los que se eliminaron signos innecesarios, se realizó la tokenización de algunas variables y se armaron nuevos datasets con los datos listos para ser usados por los modelos.

## 3) Clasificación del estado de ánimo

Para la clasificación del estado de ánimo del usuario se utilizó un **modelo de regresión logística** que clasifica el estado de ánimo del usuario en 3 categorías “Melancólico”, “Ni fu Ni fa” y “Alegre”. Este modelo fue entrenado con los datos en el dataset **sentimientos.csv** . Las frases del dataset fueron convertidos en vectores numéricos a través del embedding del modelo pre entrenado (SBERT)




“**sentence-transformers/distiluse-base-multilingual-cased-v1**” de la biblioteca **sentence-transformers**. Luego se dividió el dataset con los vectores en un set de entrenamiento con el 80% de los datos y un set de testeo con el 20% de los datos. El

set de entrenamiento se usó para entrenar el modelo de regresión logística y el de testeo para evaluar el rendimiento del modelo.

```
Precisión Regresión Logística: 0.9393939393939394
Reporte de clasificación Regresión Logística:
              precision    recall  f1-score   support

           0       0.91      0.91      0.91        11
           1       0.91      0.91      0.91        11
           2       1.00      1.00      1.00        11

    accuracy                           0.94        33
   macro avg       0.94      0.94      0.94        33
weighted avg       0.94      0.94      0.94        33
```

## 4) Sistema de recomendaciones

Para el sistema de recomendaciones se utilizó el modelo "**sentence-transformers/all-MiniLM-L12-v2**" de la biblioteca **sentence-transformers** para hacer los embeddings para aplicar técnicas de **semejanza de texto** con métodos como la **similitud de coseno** para seleccionar las mejores recomendaciones para las preferencias del usuario**.**

Para el **reconocimiento de entidades nombradas (NER)** se utilizó el modelo "**urchade/gliner_medium-v2.1**" de la librería **GLiNER.**

Para las traducciones se utilizó el modelo "**MyMemoryTranslator**" de la libreria **deep_translator.**

En el sistema de recomendaciones primero se pre calculan los embeddings de los datasets para que cuando el usuario solicite la recomendaciones los embeddings ya estén cargados y obtenga una respuesta mas rapida.

Cuando el usuario carga su preferencia, lo primero que ocurre es que se traduce la preferencia al inglés, esto se hace debido que la información en los datasets estan en ingles y es necesaria su traducción para que cuando extraigamos las entidades nombradas en la preferencia esten el mismo idioma que en los datasets.

Una vez hecha la traducción se extraen las entidades nombradas con el modelo **GLiNER.** Luego se filtra el dataset con las entidades encontradas en la preferencia del usuario para obtener resultados más precisos. Una vez filtrado el dataset se mide la **similitud del coseno** entre los embeddings del dataset y el embedding de la preferencia del usuario en inglés para devolver los 3 resultados con mayor similitud. En caso de no encontrar ningún resultado en el filtrado con las entidades nombradas se

calcula la similitud del coseno con todos los embeddings del dataset.

### 5) Interfaz de Usuario

Se utilizó la librería “ipywidgets” para agregar controles interactivos en el notebook y asi poder realizar la interacción entre los modelos y el usuario de una forma amigable, como si se tratara de una conversación con un chatbot.

Primero inicializando la conversación desde el lado del asistente, mostrando un mensaje que invita al usuario a hablar de sus sentimientos para poder disparar la clasificación del ánimo luego de obtener ese prompt.

Luego para indicarle al usuario lo que percibió el modelo y preguntarle que tipo de entretenimiento busca, “1” para libros, “2” para películas y “3” para juegos de mesa. Finalmente, reafirma la decisión del usuario y pide una temática para hacer la recomendación.

Para todas las entradas del usuario se utilizó la clase Text, que dispone de un placeholder para ayudar al usuario a responder adecuadamente. Para todas las salidas, se utilizó la clase HTML, con la descripción “Asistente:”. Se observan fácilmente cuales son los controles con los que el usuario puede interactuar.

Gracias a los botones Button, con descripción “Responder” se pudo validar las entradas del usuario para permitir una buena experiencia al utilizarlo.

## **5.** Conclusiones

Este sistema resultó eficaz para brindar recomendaciones personalizadas en situaciones donde las actividades al aire libre son limitadas, proporcionando opciones recreativas de películas, juegos de mesa y libros.

Una de las posibles mejoras, que le permitiría al sistema abarcar un rango más amplio de emociones, sería la ampliación del dataset de emociones, incluyendo más datos se enriquecería el proceso de entrenamiento del modelo de clasificación.

Se podría optimizar el sistema de recomendaciones obteniendo más información del usuario, por ejemplo, pidiendo la cantidad de personas que están buscando entretenimiento o la cantidad de tiempo del que se dispone.

Este proyecto demuestra el potencial de las técnicas de procesamiento de lenguaje natural a la hora de personalizar experiencias de entretenimiento, utilizando las bases para desarrollos futuros que pueden aprovecharse en distintos contextos y necesidades, como sugerencias de canciones para momentos específicos en un viaje, sugerencias de recetas para hacer comidas dependiendo de varios factores y más.