



UNIVERSIDAD NACIONAL DE ROSARIO
FACULTAD DE CIENCIAS EXACTAS, INGENIERÍA Y AGRIMENSURA

T.U.I.A

PROCESAMIENTO DE
LENGUAJE NATURAL

TRABAJO PRÁCTICO 1

2024

Alumno:

Pistelli, Pablo

ÍNDICE

Descripción del trabajo práctico	3
Herramientas utilizadas	3
Reconocimiento de Entidades Nombradas - NER (Named Entity Recognition)	3
Scraping	3
Generación de embeddings	4
Adaptación de idioma	4
Identificación de estado de ánimo	4
Interfaz de usuario	5
Desarrollo	6
Generación de base de datos de libros	6
Exploración y procesamiento de bases de datos	7
Búsqueda de opciones	8
<i>Interfaz de usuario</i>	8
<i>Estrategia de búsqueda</i>	9
<i>Ejemplos de uso</i>	10
Mejoras posibles	17
Conclusiones	17

DESCRIPCIÓN DEL TRABAJO PRÁCTICO

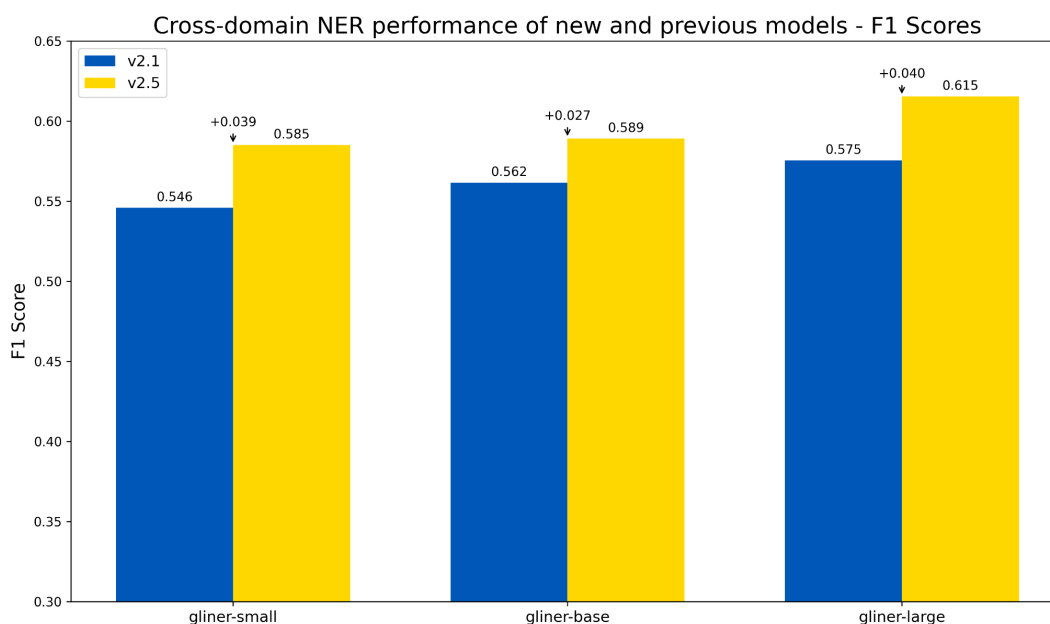
Este trabajo consiste en desarrollar un programa de Procesamiento de Lenguaje Natural que, según el estado de ánimo del usuario y sus preferencias, recomiende una película, jugar un juego de mesa o leer un libro. Para ello, se construyó un clasificador que categoriza el estado de ánimo del usuario y sugiere el conjunto de recomendaciones basada en una frase de preferencia ingresada por el usuario.

HERRAMIENTAS UTILIZADAS

RECONOCIMIENTO DE ENTIDADES NOMBRADAS - NER (NAMED ENTITY RECOGNITION)

Para identificar entidades dentro de los textos de las bases de datos se utilizó la librería **GLiNER**, provista en la teoría y utilizada en las prácticas de la materia. Para su aplicación se generaron listas de *labels* para cada tipo de contenido.

El modelo preentrenado utilizado en clase es multilinguaje. Para este trabajo en particular, al trabajar con bases de datos en inglés, es posible utilizar el modelo **gliner-community/gliner_large-v2.5** que presenta una mejora en la performance frente al utilizado en el ejemplo de clase.



SCRAPING

Para generar la base de datos de libros se extrajeron datos del sitio <https://www.gutenberg.org/>. Se utilizó la librería **BeautifulSoup**, siguiendo las técnicas aplicadas en la práctica de la Unidad 3.

GENERACIÓN DE EMBEDDINGS

Se generaron incrustaciones de los textos correspondientes a las descripciones de cada contenido para aplicar posteriormente la búsqueda semántica según los inputs de preferencias del usuario. Para esto se utilizó **SentenceTransformer** con el modelo preentrenado **msmarco-MiniLM-L-6-v3**.

ADAPTACIÓN DE IDIOMA

Todas las bases de datos se encuentran en inglés y la interacción con el usuario debe ser en español. Para cumplir con ese requerimiento se definió la siguiente estrategia:

*Input de usuario (es) → Traductor (es-en) → Procesamiento de Input (en) →
Búsqueda en bases de datos (en) → Resultados (en) → Traductor (en-es) →
Presentación al usuario (es)*

Se probaron dos métodos de traducción:

1. Modelos NMT EN-ES / ES-EN utilizando **transformers** con los modelos preentrenados **Helsinki-NLP/opus-mt-en-es** y **Helsinki-NLP/opus-mt-es-en**.
2. **GoogleTranslate** de la librería **deep-translator**.

En base a la velocidad y calidad de traducción el modo que devolvió mejores resultados fue el **Google Translate**.

IDENTIFICACIÓN DE ESTADO DE ÁNIMO

Para la identificación del estado de ánimo del usuario a partir de una frase ingresada por el mismo se utilizaron dos modelos y se compararon sus resultados.

1. Modelo 1: multilingual-uncased-sentiment
Basado en el ejemplo de la teoría de la Unidad 3 se utilizó un modelo multilingüe basado en BERT **nlptown/bert-base-multilingual-uncased-sentiment**.
Este modelo clasifica el sentimiento en una escala de 1 a 5. Las podríamos considerar: 1 = Melancólico; 3 = Neutral; 5 = Alegre.
2. Modelo 2: pysentimiento
Como alternativa, explorando el repositorio de HuggingFace, encontré un modelo entrenado con el dataset "TASS 2020 Task 2" que contiene 8,409 tweets en español etiquetados según las definidas "emociones universales" de Ekman: ira, asco, miedo, alegría, tristeza y sorpresa.

Esta herramienta presenta distintos analizadores:

- Análisis de sentimiento
- Detección de discurso de odio
- Detección de ironía
- Análisis de emoción

Para la consigna se utilizará el Análisis de sentimiento que clasifica una frase como Positiva, Negativa o Neutral con una probabilidad para cada caso.

Luego de aplicar ambos modelos de detección a una misma lista de frases, obtuve los siguientes resultados:

FRASE	MODELO 1	MODELO 2
"Hoy estoy un poco cansado."	3 ; 0.596	NEG ; 0.938
"La lluvia me pone de mal humor."	1 ; 0.510	NEG ; 0.982
"Me gustan los días nublados."	4 ; 0.540	POS ; 0.879
"Hoy es un buen día para quedarse en casa tomando mates."	4 ; 0.426	POS ; 0.934
"No me importa qué hagamos hoy."	2 ; 0.339	NEU ; 0.725
"Las tormentas me asustan."	2 ; 0.400	NEG ; 0.953

Para el desarrollo del trabajo se utilizará el **Modelo 2: pysentimiento** ya que se adapta mejor a la consigna de clasificar el estado de ánimo en 3 categorías (Positivo, Neutral, Negativo). Por otro lado, comparando los resultados de análisis sobre las mismas frases, considero más acertados los del segundo modelo.

INTERFAZ DE USUARIO

Se utilizó la librería **ipywidgets** para generar los campos de texto y selección de opciones del usuario.

DESARROLLO

GENERACIÓN DE BASE DE DATOS DE LIBROS

NOTEBOOK: NLP_TP1_DATA.IPYNB

Para generar la base de datos de libros se utilizaron técnicas de scraping sobre el sitio <https://www.gutenberg.org/>. Se decidió extraer el listado de los 1000 libros más populares en los últimos 30 días.

La web presenta un listado de 1000 títulos, cada uno contiene un hipervínculo a la página individual del libro donde se encuentra la información que se desea extraer.

El primer paso fue generar un listado de las 1000 urls. Se identificaron las secciones 'ol'. Cada una corresponde a un Top 1000 diferente:

- [0] Top 1000 EBooks yesterday
- [1] Top 1000 Authors yesterday
- [2] Top 1000 EBooks last 7 days
- [3] Top 1000 Authors last 7 days
- [4] Top 1000 EBooks last 30 days
- [5] Top 1000 Authors last 30 days

Para nuestra base de datos elegiremos el **Top 1000 Eboks last 30 days**.

Dentro de la sección elegida, se extrajeron las secciones 'li', que corresponden a cada título de la lista.

```
▼<li>
  ::marker
  <a href="/ebooks/84">Frankenstein; Or, The Modern Prometheus by Mary Wollstonecraft Shelley (174761)</a> == $0
</li>
```

El texto a extraer en la imagen de ejemplo es '/ebooks/84'. Para esto se aplicó en cada línea:

```
# Genero una lista para acceder a la url de cada libro
url_dominio = 'https://www.gutenberg.org'
urls_libros = []

for titulo in titulos_top1000:
    # En href se encuentra la subcarpeta y página con la información de cada
    titulo
    urls_libros.append(url_dominio + titulo.find('a').attrs['href'])
```

Obteniendo una lista del tipo:

```
urls_libros[0:5]

['https://www.gutenberg.org/ebooks/84',
 'https://www.gutenberg.org/ebooks/1342',
 'https://www.gutenberg.org/ebooks/1513',
 'https://www.gutenberg.org/ebooks/2701',
 'https://www.gutenberg.org/ebooks/25344',
 'https://www.gutenberg.org/ebooks/145',]
```

Luego, mediante la iteración de esa lista de 1000 urls, se aplicó la misma técnica para obtener los datos de interés y volcarlos en un dataframe. Cada fila del dataframe corresponde a una url con los datos del título en cuestión.

Cada página de cada libro tiene una tabla con información, pero no todos los libros tienen la misma información disponible ni ordenada de la misma manera.

Para generar la base de datos se extrajo:

- Title
- Author
- Summary
- Subjects
- URL

Finalmente, se guardó el dataframe en un archivo csv para utilizarlo junto a las bases de datos provistas.

EXPLORACIÓN Y PROCESAMIENTO DE BASES DE DATOS

NOTEBOOK: NLP_TP1_DATA.IPYNB

Una vez listas las tres bases de datos, se realizó una exploración de cada una para identificar la información relevante para el proyecto.

El primer procesamiento que se realizó fue el reconocimiento de entidades nombradas utilizando las herramientas mencionadas. Esta técnica se aplicó sobre el corpus de las descripciones o resúmenes de cada libro, película o juego. Al tratarse de diferentes tipos de contenidos, se definieron diferentes labels para cada uno:

```
# Lista de etiquetas que el modelo intentará encontrar en el texto
labels_pelicula = ["movie", "person", "book", "location", "date", "actor",
"character", ""]
labels_juego = ["game", "mechanic", "character", "piece", "theme", "date",
"author", "action", ""]
labels_libro = ["person", "book", "location", "date", "character", ""]
```

Las entidades reconocidas en cada caso se guardaron en una nueva columna dentro de la base de datos inicial.

Luego, para la búsqueda semántica se generaron los embeddings de dichos textos y se guardan en archivos .npy. Esto es para no tener que procesarlos en cada ejecución del proyecto.

Tanto las bases de datos extendidas y los embeddings se exportaron y guardaron en la carpeta 'Data' del proyecto.

BÚSQUEDA DE OPCIONES

NOTEBOOK: NLP_TP1_RECOMENDACIONES.IPYNB

INTERFAZ DE USUARIO

Para la utilización del sistema de recomendaciones se generó una pequeña interfaz de usuario que consiste en los siguientes campos:

¿Qué estás pensando?

Respuesta...

¿Cuáles son tus preferencias?

Respuesta...

¿Qué tipo de contenido quieres que te recomiende?

Libros

Películas

Juegos

¿Cuántas opciones quieres que busque?

1



Buscar recomendaciones

¿Qué estás pensando?

Sobre lo ingresado en este campo se aplica el reconocimiento de emociones.

¿Cuáles son tus preferencias?

Sobre lo ingresado en este campo se aplica el reconocimiento de entidades nombradas y se generan embeddings para búsqueda semántica.

¿Qué tipo de contenido quieres que te recomiende?

Según la selección se aplica la búsqueda a la base de datos correspondiente.

¿Cuántas opciones quieres que busque?

Selección entre 1 y 5 resultados de búsqueda.

ESTRATEGIA DE BÚSQUEDA

Según lo explorado en las distintas bases de datos, se define la siguiente estrategia de búsqueda:

Filtrar bases de datos

El primer análisis consiste en seleccionar parte de la base de datos según un criterio determinado:

- **Películas:** Según el reconocimiento de emociones, primero se filtrará la base de datos por género:
 - POSITIVO: 'Action', 'Adventure', 'Animation', 'Comedy', 'Romance', 'Musical'
 - NEUTRAL: 'Mystery', 'Sci-Fi', 'Family', 'Fantasy', 'Music', 'Crime', 'War', 'Sport'
 - NEGATIVO: 'Horror', 'Thriller', 'Drama', 'Biography', 'History', 'Western'
- **Juegos:** Se podría aplicar un criterio similar al de los géneros de las películas, pero al contar con muchos más géneros, el primer filtro se realizará buscando coincidencias en entidades nombradas. En el caso de que no se encuentre ninguna coincidencia, se utilizará la base de datos completa.
- **Libros:** La lista de temas también contiene muchas variantes, por lo que se aplicará el mismo criterio que en los juegos.

Para cada caso se obtiene un dataframe filtrado con los contenidos que coinciden con la búsqueda y los embeddings correspondientes a cada uno de ellos.

Los embeddings son extraídos del archivo generado en el procesamiento de datos según el índice.

Búsqueda semántica

El siguiente paso es realizar una búsqueda semántica asimétrica sobre las descripciones/resúmenes de las películas, libros y juegos de las bases de datos ya filtradas. Esto se debe a que se cumplen ambas características mencionadas en la teoría de este tipo de búsquedas:

- Diferencia significativa en la longitud de los textos
- No intercambiabilidad entre la consulta y el corpus

Para esto se generan los embeddings de las preferencias y se realiza la búsqueda de los mejores resultados según la cantidad ingresada por el usuario dentro de los embeddings filtrados en el primer paso.

Presentación de los resultados

Una vez obtenidos los resultados se extrae la información original del dataframe y se presenta por consola al usuario. La información textual (títulos, géneros, descripciones, temas, etc.) se traduce al español utilizando las funciones desarrolladas anteriormente.

EJEMPLOS DE USO

¿Qué estás pensando?

Hoy es un gran día!

¿Cuáles son tus preferencias?

Me gustan las aventuras de personajes graciosos en situaciones peligrosas.

¿Qué tipo de contenido quieres que te recomiende?

Libros
Películas
Juegos

¿Cuántas opciones quieres que busque?

2

Buscar recomendaciones

Veamos que puedo encontrar para recomendarte...

Contenido seleccionado: Películas

Cantidad seleccionada: 2

Veo que estas con una actitud positiva!

Respuesta 1 (Similitud: 0.4119)

Título: Coralina

Título original: Coraline

Género: Animación, Familia, Fantasía

Descripción: Una chica aventurera encuentra otro mundo que es una versión extrañamente idealizada de su frustrante hogar, pero tiene secretos siniestros.

Director: Henry Selick

Actores: Dakota Fanning, Teri Hatcher, John Hodgman, Jennifer Saunders

Año: 2009

Duración: 100

Rating: 7.7

Respuesta 2 (Similitud: 0.4031)

Título: 2012

Título original: 2012

Género: Acción, Aventura, Ciencia ficción

Descripción: Un escritor frustrado lucha por mantener viva a su familia cuando una serie de catástrofes globales amenaza con aniquilar a la humanidad.

Director: Roland Emmerich

Actores: John Cusack, Thandie Newton, Chiwetel Ejiofor, Amanda Peet

Año: 2009

Duración: 158

Rating: 5.8

¿Qué estás pensando?

Me gustan mucho los días nublados

¿Cuáles son tus preferencias?

Me interesan las historias sobre la época de la primera guerra mundial y cómo vivía

¿Qué tipo de contenido quieres que te recomiende?

Libros

Películas

Juegos

¿Cuántas opciones quieres que busque?

3



Buscar recomendaciones

Veamos que puedo encontrar para recomendarte...

Contenido seleccionado: Libros

Cantidad seleccionada: 3

Respuesta 1 (Similitud: 0.3372)

Título: Historia actual, vol. VIII, n.º 3, junio de 1918
Revista mensual del New York Times

Título original: Current History, Vol. VIII, No. 3, June 1918
A Monthly Magazine of the New York Times

Autor: Various

Resumen: "Current History, Vol. VIII, No. 3, June 1918" de Various es una publicación de revista histórica escrita a principios del siglo XX. Detalla los acontecimientos contemporáneos durante la Primera Guerra Mundial, centrándose en los aspectos militares, políticos y sociales de la época, en particular las batallas y estrategias en el frente occidental. El contenido incluye informes de corresponsales y análisis de situaciones de combate, que reflejan las experiencias y los sacrificios de los soldados durante este tumultuoso período. La apertura de este volumen ofrece una instantánea de la situación en tiempos de guerra en mayo de 1918, destacando la relativa calma en el frente occidental después de intensos combates. Describe movimientos y enfrentamientos militares cruciales, como las batallas en Picardía y Flandes, la llegada de tropas estadounidenses a Francia y los incansables intentos alemanes de romper las líneas aliadas. Además, analiza los acontecimientos políticos clave, incluida la renovación de las alianzas entre Alemania y Austria y el impacto significativo de las misiones laborales estadounidenses en Europa, pintando un panorama completo y dinámico de la vida y el conflicto durante la Primera Guerra Mundial. (Este es un resumen generado automáticamente).

Temas: ['Guerra Mundial, 1914-1918']

URL: <https://www.gutenberg.org/ebooks/41479>

Respuesta 2 (Similitud: 0.3150)

Título: Mi vida – Volumen 1

Título original: My Life – Volume 1

Autor: Wagner, Richard, 1813-1883

Resumen: "Mi vida - Volumen 1" de Richard Wagner es una obra autobiográfica escrita a mediados del siglo XIX. Este volumen detalla los primeros años y las experiencias formativas del famoso compositor, desde su infancia en Leipzig hasta sus primeros esfuerzos profesionales alrededor de 1842. El libro reflexiona sobre los antecedentes familiares de Wagner, su introducción a la música y las influencias que dieron forma a su trayectoria artística. El comienzo de este relato autobiográfico comienza con el nacimiento de Wagner y las consecuencias inmediatas de la muerte de su padre, que dejó a su familia en circunstancias difíciles. Describe el papel fundamental de su padrastro en su crianza y educación, así como el vibrante entorno cultural en el que estuvo inmerso, incluidos sus primeros encuentros con el teatro y la música. A lo largo de la narración, Wagner relata sus recuerdos de infancia, las dificultades con la educación y el florecimiento de su talento musical, que en última instancia prepara el escenario para su futura carrera como compositor. El lector se adentra en las emociones y aspiraciones de un joven artista a punto de descubrir su vocación, con acontecimientos, relaciones y sentimientos significativos que presagian la extraordinaria vida que llevaría. (Este es un resumen generado automáticamente.)

Temas: ['Wagner, Richard, 1813-1883', 'Compositores -- Alemania -- Biografía']

URL: <https://www.gutenberg.org/ebooks/5197>

Respuesta 3 (Similitud: 0.2849)

Título: La crónica anglosajona

Título original: The Anglo-Saxon Chronicle

Autor: nan

Resumen: "La Crónica anglosajona" de J. Ingram y J. A. Giles es un relato histórico compilado originalmente durante el reinado del rey Alfredo el Grande, alrededor de finales del siglo IX, y continuado hasta el siglo XII por varios escribas anónimos. Esta crónica sirve como registro vital de la historia temprana de Inglaterra, documentando eventos significativos desde la llegada de los sajones hasta la conquista normanda, ilustrando el desarrollo del estado inglés y su gente. La apertura de la crónica presenta el contexto geográfico e histórico de Gran Bretaña, describiendo su longitud y anchura, y destacando a los primeros habitantes clave, incluidos los britanos, los pictos y los escoceses. Relata la invasión romana liderada por Julio César, las batallas posteriores libradas por estos primeros pueblos y la migración de diferentes grupos a la isla. Esta parte establece una base para comprender las complejidades de la composición étnica y cultural temprana de Gran Bretaña, que condujo al establecimiento final de los reinos anglosajones. La narración está estructurada cronológicamente, con los acontecimientos más importantes marcados por los nombres de los líderes y los sucesos notables, lo que la convierte en una fuente única y esencial para cualquier persona interesada en la historia medieval temprana de Inglaterra. (Este es un resumen generado automáticamente).

Temas: ['Gran Bretaña -- Historia -- Período anglosajón, 449-1066 -- Fuentes', 'Anglosajones -- Obras tempranas hasta 1800']

URL: <https://www.gutenberg.org/ebooks/657>

¿Qué estás pensando?

No me importa qué hagamos hoy

¿Cuáles son tus preferencias?

Me gustan las aventuras medievales y los acertijos.

¿Qué tipo de contenido quieres que te recomiende?

Libros
Películas
Juegos

¿Cuántas opciones quieres que busque?

4



Buscar recomendaciones

Veamos que puedo encontrar para recomendarte...

Contenido seleccionado: Juegos

Cantidad seleccionada: 4

Respuesta 1 (Similitud: 0.4898)

Nombre: Proyecto L

Nombre original: Project L

Rating promedio: 7.49

Descripción: ¡Construye piezas, desarrolla un motor, perfecciona tu estrategia y gana el juego!

Project L es un juego de combinación de piezas de ritmo rápido con rompecabezas 3D de tres capas y hermosas piezas de acrílico. Desafía a tus amigos a un juego de diseño simple pero con una jugabilidad intrincada que deja una impresión duradera.

La base del juego radica en usar tus piezas para completar rompecabezas. Comenzando con solo dos piezas básicas, usas tres acciones en cada turno para desarrollar un motor poderoso. Con más piezas de varios tipos, puedes completar de manera eficiente incluso los rompecabezas más difíciles. Los rompecabezas que completes te otorgarán puntos o nuevas piezas para alimentar aún más tu motor. ¿Puedes ser más astuto que tus oponentes?

—descripción del editor

Desarrollador: ['Michal Mikeš', 'Jan Soukal', 'Adam Španěl']

Año: 2020

Categorías: ['Estrategia abstracta', 'Rompecabezas', 'Tema de videojuego']

Mecánicas: ['Puntos de acción', 'Borrador abierto', 'Creación de patrones', 'Cola de recursos', 'Colocación de mosaicos']

Tiempo[min]: 20 a 40

Número de jugadores: 1 a 4

Respuesta 2 (Similitud: 0.3898)

Nombre: Mansiones de la locura: Segunda edición

Nombre original: Mansions of Madness: Second Edition

Rating promedio: 7.95

Descripción: Mansions of Madness: Second Edition es un juego de mesa de terror y misterio totalmente cooperativo, basado en una aplicación, para uno a cinco jugadores, que se desarrolla en el mismo universo que Eldritch Horror y Elder Sign. Deja que la aplicación inmersiva te guíe por las calles veladas de Innsmouth y los pasillos embrujados de las mansiones malditas de Arkham mientras buscas respuestas y un respiro. Ocho valientes investigadores están listos para enfrentar cuatro escenarios de miedo y misterio, recolectando armas, herramientas e información, resolviendo acertijos complejos y luchando contra monstruos, locura y muerte. Abre la puerta y entra en estas espeluznantes Mansions of Madness: Second Edition. Hará falta algo más que sobrevivir para conquistar los males que aterrorizan a esta ciudad.

Desarrollador: ['Nikki Valens']

Año: 2016

Categorías: ['Aventura', 'Exploración', 'Lucha', 'Terror', 'Miniaturas', 'Asesinato/Misterio']

Mecánicas: ['Movimiento de área', 'Juego cooperativo', 'Lanzamiento de dados', 'Gestión de manos', 'Adición de mapas', 'Tablero modular']

Tiempo[min]: 120 a 180

Número de jugadores: 1 a 5

Respuesta 3 (Similitud: 0.3066)

Nombre: Descenso: Viajes en la oscuridad

Nombre original: Descent: Journeys in the Dark

Rating promedio: 7.28

Descripción: Descent: Journeys in the Dark es un juego de "uno contra muchos" en el que hasta cuatro jugadores elegirán personajes con una amplia variedad de habilidades y capacidades innatas para ser los héroes que explorarán las mazmorras en busca de tesoros y aventuras, mientras que un jugador asumirá el papel del Overlord y controlará las numerosas trampas, acertijos y monstruos de la mazmorra.

El objetivo de los héroes será conquistar la mazmorra de forma cooperativa, apoderarse de sus numerosos tesoros y lograr otros objetivos establecidos por el escenario. Si los héroes cooperan y logran sus objetivos, todos ganarán. El objetivo del Overlord es simplemente utilizar todos los medios a su disposición, desde trampas mortales hasta monstruos feroces, para matar a los héroes. Cada héroe tiene un cierto valor de Puntos de Conquista para el grupo y si se pierden demasiados Puntos de Conquista por la muerte del héroe, el grupo pierde y el Overlord gana.

Reimplementado por:

Descent: Journeys in the Dark (Segunda edición) (2012)

Desarrollador: ['Kevin Wilson']

Año: 2005

Categorías: ['Aventura', 'Exploración', 'Fantasía', 'Lucha', 'Miniaturas']

Mecánicas: ['Lanzamiento de dados', 'Movimiento en cuadrícula', 'Tablero modular', 'Juego de rol', 'Juego en equipo', 'Poderes de jugador variables']

Tiempo[min]: 120 a 240

Número de jugadores: 2 a 5

Respuesta 4 (Similitud: 0.2972)

Nombre: Exit: El juego - La cabaña abandonada

Nombre original: Exit: The Game - The Abandoned Cabin

Rating promedio: 7.42

Descripción: Todos querían usar la cabaña solo como refugio por la noche, pero cuando llega la mañana, la puerta está asegurada con una cerradura de combinación, y nadie conoce la combinación de números que les permitirá salir. Las ventanas también están enrejadas. Un enigmático dial giratorio con códigos y un misterioso libro es todo lo que tienes para seguir. ¿Podrás escapar de esta cabaña abandonada?

En Exit: The Game - The Abandoned Cabin, los jugadores deben usar su espíritu de equipo, creatividad y poderes de deducción para descifrar códigos, resolver acertijos, recolectar objetos y ganar su libertad poco a poco.

Desarrollador: ['Inka Brand', 'Markus Brand']

Año: 2016

Categorías: ['Deducción', 'Rompecabezas', 'Tiempo real']

Mecánicas: ['Juego cooperativo']

Tiempo[min]: 60 a 120

Número de jugadores: 1 a 6

MEJORAS POSIBLES

En base a lo aprendido durante el desarrollo de este proyecto identifiqué algunos puntos que podría mejorar:

1. Mejorar la modularidad del código. Al llegar a la etapa de búsquedas en las bases de datos noté que es posible separar los diferentes procesos de la búsqueda en funciones para mejorar el código y poder reutilizarlo sobre nuevas bases de datos. Por ejemplo, generar una única función de salida que reciba como parámetros los ítems que se deben mostrar de cada tipo de contenido.
2. La primera búsqueda con criterios de filtrado es individual para cada tipo de contenido (libro, película o juego). Sería posible también aplicar más de un filtrado antes de realizar la búsqueda semántica.
3. Mejorar las bases de datos. El preprocesamiento de las bases de datos fue simplemente reconocer entidades y generar embeddings. Se podría analizar más en detalle el contenido para eliminar caracteres no deseados o también probar aplicando técnicas de POS tagging.
4. Mejorar la interfaz de usuario. Se podrían explorar más herramientas de la librería ipywidgets para mejorar la interacción y el ingreso de datos. También sería más amigable al usuario generar una única función *'mostrar_interfaz()'* para ejecutar el modelo de recomendación y devolver los resultados en otro elemento que no sea la consola del notebook.

CONCLUSIONES

Los modelos de procesamiento de lenguaje natural utilizados en el trabajo resultaron simples de aplicar y obtuve resultados satisfactorios en la búsqueda de recomendaciones.

Considero que un paso importante fue el procesamiento de las bases de datos por separado y su almacenamiento fuera del modelo. La generación de embeddings también es un proceso que requiere un tiempo considerable de ejecución, por lo que almacenarlo junto a las bases de datos resulta muy conveniente.

Si bien utilizamos modelos preentrenados muy amplios en el desarrollo de la materia, una buena práctica fue explorar los modelos disponibles en HuggingFace para encontrar nuevos más potentes, o bien, modelos que se adapten mejor a la aplicación que estamos llevando a cabo.