

Tecnicatura Universitaria en Inteligencia Artificial

Procesamiento del Lenguaje Natural

**Trabajo Práctico 1**

### 

### 

### 

### Integrantes:

* Herrera Morena - (H-1187/8)
* Zorzolo Rubio Juana - (Z-1217/3)

Profesores:

* Juan Pablo Manson
* Alan Geary

Fecha de entrega: 06/11/24

**ÍNDICE**

[Resumen 3](#_qcvrvxm4s4u7)

[Introducción 3](#_sbcl7gfvyxhj)

[Objetivos Específicos 3](#_j9xb7ekd4xkp)

[Metodología 4](#_wv6qzbwbseuz)

[Desarrollo/Implementación 5](#_4lrud2z7uoyd)

[Resultados 7](#_tp9yqim88ir6)

[Conclusiones 8](#_tm0uizq3k5k3)

[Referencias 9](#_o8rq34ibepum)

[Anexos 9](#_70mw3fl12r8e)

## 

## 

## **Resumen**

Este trabajo práctico presenta el desarrollo de un clasificador de recomendaciones recreativas basado en el procesamiento de lenguaje natural (NLP). El objetivo es ofrecer opciones de entretenimiento (como películas, juegos de mesa y libros) para días de lluvia, según una frase de preferencia ingresada por el usuario.

El sistema clasifica el estado de ánimo en categorías predefinidas, utilizando un modelo de clasificación supervisado para comprender el contexto emocional del usuario y métodos de embeddings y NER para calcular la similitud semántica entre la frase ingresada y una base de datos de opciones recreativas.

Los datos utilizados provienen de fuentes como bases de datos de películas y juegos y del Proyecto Gutenberg con técnicas de web scraping para extraer la información de libros. Los resultados muestran recomendaciones personalizadas y coherentes, logrando el objetivo de crear un sistema útil y adaptado al idioma español.

En el futuro, se propone optimizar el sistema para mejores resultados y menor uso de recursos. Además, explorar nuevas fuentes de datos para ampliar el catálogo de recomendaciones.

## **Introducción**

**Contexto y Justificación**El entretenimiento ganó relevancia en situaciones donde las actividades al aire libre son limitadas por condiciones climáticas adversas. En este trabajo, se desarrolla una solución que NLP, analiza el estado de ánimo del usuario y proporciona recomendaciones recreativas personalizadas basadas en las preferencias ingresadas. Esta herramienta puede ser de utilidad durante días de lluvia en vacaciones, donde las personas buscan alternativas para pasar el tiempo dentro de sus alojamientos. El uso de NLP permite una interacción natural en español, facilitando una experiencia más accesible para usuarios hispanohablantes.

**Propósito del Proyecto**El propósito de este proyecto es construir un sistema de recomendación que pueda comprender las preferencias del usuario y proponer opciones de entretenimiento (películas, juegos de mesa y libros). Para lograrlo, el sistema utiliza un clasificador supervisado que identifica el estado de ánimo, combinado con técnicas de embeddings y reconocimiento de entidades nombradas para optimizar la búsqueda y relevancia de las recomendaciones.

### **Objetivos Específicos**

1. **Desarrollar un clasificador de estados de ánimo** que categorice el estado emocional del usuario a partir de una frase ingresada en español, con categorías como "Alegre", "Melancólico" o "Ni fu ni fa".
2. **Implementar una técnica de procesamiento semántico** mediante embeddings y reconocimiento de entidades nombradas para calcular la similitud entre la preferencia ingresada por el usuario y las descripciones de las opciones de entretenimiento.
3. **Construir un sistema de scraping** para extraer información sobre libros desde el Proyecto Gutenberg y estructurarla en un dataset de fácil acceso.
4. **Integrar y estructurar los datos de las distintas fuentes** (juegos de mesa, películas y libros) para permitir recomendaciones adaptadas a los gustos del usuario.
5. **Proporcionar un sistema de recomendaciones en español** que logre una interacción natural con el usuario hispanohablante, facilitando el entendimiento y la usabilidad de la herramienta.

## **Metodología**

**1. Fuentes de Datos**

* **Juegos de Mesa**: La base de datos de juegos de mesa utilizada es “bgg\_database.csv,” que contiene información detallada sobre diferentes tipos de juegos, proporcionando una amplia gama de opciones.
* **Películas**: Los datos sobre películas se obtuvieron de “IMDB-Movie-Data.csv,” que contiene datos sobre películas de diversas temáticas y géneros.
* **Libros**: Para los libros, se realizó web scraping en el sitio Proyecto Gutenberg (<https://www.gutenberg.org>), extrayendo datos de los 1000 libros más populares. El proceso se realizó con BeautifulSoup para analizar el HTML, y las listas de libros fueron filtradas y estructuradas en un DataFrame para su posterior análisis.

**2. Técnicas de Procesamiento de Lenguaje Natural**

* **Clasificación Supervisada del Estado de Ánimo**: Se utilizó un modelo de clasificación supervisada, implementado con el paquete Hugging Face Transformers y un modelo preentrenado de BERT, especificamente nlptown/bert-base-multilingual-uncased-sentiment. Este clasificador fue diseñado para identificar el estado de ánimo en frases en español, categorizando el estado emocional en etiquetas del 1 al 5. Luego en base a la etiqueta, se clasifica el sentimiento del usuario como "Alegre", "Melancólico", o "Ni fu ni fa". Además, se utilizó un pipeline de clasificación para simplificar el proceso de inferencia. El resultado es el sentimiento clasificado.
* Traducción: Debido a que la frase ingresada está en español pero los datas en inglés, se utilizó el modelo de traducción automática NMT con su tokenizador. En específico el modelo es Helsinki-NLP/opus-mt-es-en, el cual está diseñado para traducir textos de español a inglés.
* **Embeddings y Similitud Semántica**: Se implementó un modelo de embeddings usando SentenceTransformer, específicamente el modelo paraphrase-multilingual-MiniLM-L12-v2. Este convierte tanto las descripciones como la preferencia ingresada en vectores que capturan el significado semántico. Luego, se calculó la similitud semántica utilizando la similitud del coseno entre la frase de preferencia y cada descripción. Este enfoque permite una visión basada en el contexto.
* **Reconocimiento de Entidades Nombradas (NER)**: Para identificar entidades específicas (personas, años, locaciones, etc.) dentro de la frase de preferencia y las descripciones de las opciones recreativas, se utilizó un modelo de reconocimiento de entidades nombradas (NER) gliner\_multi-v2.1, cargado a través de la biblioteca GLINER. Este proceso optimiza la búsqueda de coincidencias y asegura que las recomendaciones sean coherentes con los intereses específicos del usuario.

**3. Herramientas y Tecnologías Empleadas**

* **Software**: Python, junto con bibliotecas como pandas, numpy, BeautifulSoup, requests, transformers de Hugging Face y Gliner.
* **Hardware**: El desarrollo se llevó a cabo en Google Colab, aprovechando la infraestructura en la nube para la carga y el procesamiento de modelos pesados de NLP.

**4. Pasos del Proceso de Implementación y Cumplimiento de Objetivos**

* **Carga y Exploración de Datos**: Se comenzó con la carga de los datasets, inspeccionando su estructura y realizando ajustes necesarios en el preprocesamiento, como cambio en el nombre de las variables y eliminación de columnas innecesarias para el sistema.
* **Web Scraping y Preparación del Dataset de Libros**: Se extrajo y procesó la lista de libros populares desde el Proyecto Gutenberg. Esto incluyó la limpieza y formateo de los datos para su inclusión en el sistema de recomendaciones.
* **Construcción del Clasificador de Estado de Ánimo**: Se implementó un clasificador utilizando el modelo BERT, evaluando su desempeño en frases de prueba en español para asegurar una clasificación de alta precisión. En base a estas pruebas, se eligió el rango numérico para categorizar el estado de ánimo (alegre, melancólico o ni fu ni fa).
* **Implementación de la Similaridad Semántica y NER**: Con el estado de ánimo clasificado y la frase de preferencia ingresada, se calcularon las similitudes entre las frases del usuario y las descripciones de las opciones de entretenimiento, complementado con NER para capturar entidades clave.
* **Validación y Pruebas**: El sistema fue sometido a pruebas con diferentes frases de entrada para evaluar la efectividad y coherencia de las recomendaciones.

## **Desarrollo/Implementación**

**1. Preparación y Preprocesamiento de Datos**

* **Exploración Inicial de los Datasets**: En primer lugar, cada dataset fue cargado y explorado utilizando pandas, lo cual permitió identificar estructuras y tipos de datos.

En particular, el dataset de libros, se generó a partir de un script de web scraping en el Proyecto Gutenberg. Utilizando BeautifulSoup, el proceso comenzó con la obtención de la lista de los 1000 libros más populares. A través de la búsqueda de enlaces específicos, se identificaron aquellos que contenían "/ebooks/" seguido de un número, garantizando así la validez de los libros seleccionados. De cada libro se extrajo información clave y se realizó una limpieza básica para asegurar un mejor procesamiento de los datos.

Para evitar sobrecargar el servidor del Proyecto Gutenberg, el proceso de web scraping incluyó un mecanismo de control de flujo de una pausa de 0.01 segundos entre cada solicitud. Una vez completada la recolección de los primeros 1000 libros, los datos obtenidos se almacenaron en un DataFrame.

* **Preprocesamiento de los Datasets y texto:** Para mayor facilidad, se renombraron columnas importantes dentro de los tres datas y eliminaron las variables que no se precisaban para la recomendación. Además, para asegurar consistencia, se realizaron transformaciones en el texto, como por ejemplo convertirlo a minúsculas, eliminar acentos y caracteres no deseados y reemplazar cualquier signo que no sea alfanumérico por un espacio. Este paso es crucial para mejorar la calidad de la entrada y la efectividad de los modelos en las tareas de NER y embeddings.

**2. Clasificador del Estado de Ánimo**

* **Modelo de Clasificación Supervisada**: Para la clasificación del estado de ánimo en categorías se utilizó un modelo BERT. La entrada es una frase breve del usuario que expresa su estado emocional actual, la misma es procesada y clasificada por el modelo para obtener una etiqueta correspondiente entre 1 y 5. De esta manera se la clasifica y define al usuario como "Alegre", "Melancólico", y "Ni fu ni fa".

**3. Traducción y procesamiento de preferencia**

A la preferencia ingresada por el usuario se la traduce con NMT y luego se aplican las técnicas de parseo, embedding y NER. Dentro del código, se elige ir mostrando la frase ingresada, la frase limpia y su traducción. Posteriormente, se obtienen las entidades de la traducción y se calculan los embeddings, utilizando los modelos y funciones previamente cargados y definidos.

La función retorna los embeddings de la preferencia junto con las entidades encontradas para la futura búsqueda de similitudes con las descripciones de los datas.

**3. Embeddings y Comparación Semántica**

* **Generación de Embeddings**: Una vez determinado el estado de ánimo, el usuario ingresa una frase de preferencia. A esta frase se la traduce y se aplica un modelo de embeddings, obteniendo representaciones vectoriales para la frase del usuario.
* **Cálculo de Similitud**: Con estos embeddings, se calcula la similitud entre la preferencia del usuario y los elementos en las bases de datos de películas, juegos de mesa y libros. La similitud se mide con la distancia coseno, de modo que las recomendaciones con mayor valor de similitud son las que el sistema prioriza.

**4. Reconocimiento de Entidades Nombradas (NER)**

* **Identificación de Entidades Clave**: Utilizando un modelo NER, el sistema detecta entidades en la frase de preferencia (por ejemplo, “estados unidos” “locación” o “19/2/1983” “date”). Estas entidades se consideran al momento de realizar la búsqueda de recomendaciones, filtrando y priorizando elementos que contengan entidades matcheadas.

**5. Generación de Recomendaciones**

* **Filtrado de recomendaciones:** Para encontrar las mejores tres opciones de películas, juegos de mesa y libros se integran los dos enfoques elegidos: la similitud entre los embeddings y las coincidencias de entidades.

Iterando sobre las filas del data que recibe como parámetro, se extrae su embedding y se calcula la distancia que tiene con el embedding de la preferencia, utilizando la similitud del coseno. L

Esta fórmula mide qué tan similares son dos vectores: un valor cercano a 1 indica alta similitud.

A continuación, la función compara las entidades que se encuentran en la preferencia del usuario con las que están presentes en la descripción del elemento. Este paso se realiza mediante la búsqueda de coincidencias, contando cuántas entidades de la preferencia aparecen en la descripción del elemento. Las entidades podrían ser, por ejemplo, nombres de personas, lugares o conceptos importantes dentro del contexto.

Para cada elemento, se construye un diccionario que contiene el nombre del elemento, la cantidad de coincidencias de entidades y la similitud entre los embeddings. Si existen columnas adicionales en el DataFrame, como el nombre del autor o director, o un enlace al contenido, estos también se agregan al diccionario para enriquecer la recomendación.

Una vez procesadas todas las filas, la función crea un nuevo DataFrame con los resultados y lo ordena. Primero, ordena los elementos por la cantidad de coincidencias de entidades, y en caso de empate, lo hace por la similitud de los embeddings. Luego selecciona los tres elementos mejor posicionados en base a estos criterios.

Finalmente, se seleccionan las columnas de interés (como el nombre del elemento, el autor, el enlace, las coincidencias y la similitud) para crear un DataFrame de resultados que es retornado. Este DataFrame contiene las tres mejores recomendaciones personalizadas según la preferencia del usuario.

Este enfoque asegura que las recomendaciones no solo sean relevantes por su contenido similar, sino también por las entidades que el usuario considera importantes, lo que mejora la precisión y la personalización de las sugerencias.

* **Integración de los Resultados**: Finalmente, el sistema genera un conjunto de recomendaciones en función a la similitud semántica y la coincidencia de entidades de la frase de preferencia con la descripción. De esta manera, se proporciona una lista personalizada de 3 películas, juegos de mesa y libros.

## **Resultados**

**1. Desempeño del Clasificador de Estado de Ánimo**

* El modelo de clasificación de estado de ánimo mostró un desempeño satisfactorio en la identificación de emociones, con un alto nivel de precisión en la clasificación de categorías "Alegre," "Melancólico," y "Ni fu ni fa”.
* Se llevaron a cabo pruebas adicionales utilizando un conjunto de frases en español para asegurar que el modelo BERT pudiera interpretar adecuadamente las emociones en el idioma, lo que reforzó su precisión en el contexto de uso.

**2. Evaluación de Recomendaciones Basadas en Similitud Semántica**

* Para evaluar la efectividad del sistema de recomendación, se realizaron pruebas con frases de preferencia relacionadas a temas específicos. Por ejemplo, una frase como "una historia de amor en la selva" resultó en recomendaciones adecuadas de películas románticas y libros de aventura que contienen temas similares.
* Los resultados fueron presentados al usuario en orden de relevancia, calculada según la similitud semántica. En la mayoría de los casos, las recomendaciones más relevantes lograron captar el contexto de la preferencia ingresada, presentando opciones alineadas con las expectativas del usuario.

**3. Análisis de Resultados con NER (Reconocimiento de Entidades Nombradas)**

* Las pruebas que incluyeron términos específicos o nombres de personajes en las frases de preferencia mostraron que el modelo de NER fue efectivo en identificar palabras clave, mejorando la calidad de las recomendaciones
* En los casos donde el modelo NER no detectó coincidencias exactas, las recomendaciones aún se mantuvieron coherentes gracias al cálculo de similitud semántica mostrando resultados más precisos en general.

**4. Interacción con el Usuario**

* El sistema permite al usuario ingresar su estado de ánimo y una preferencia temática para generar recomendaciones personalizadas. Después de verificar que ambos campos estén completos, se realiza el análisis de sentimientos y el reconocimiento de entidades nombradas (NER) para identificar emociones y temas clave.
* En caso de campos incompletos, el sistema muestra un mensaje de error claro, ayudando al usuario a ajustar su entrada. Finalmente, se presentan las tres recomendaciones principales en cada categoría (juegos, películas, libros), ordenadas por relevancia según el estado de ánimo y la preferencia indicada, logrando una interacción intuitiva y efectiva.

## **Conclusiones**

**1. Evaluación del Cumplimiento de los Objetivos**

* El sistema logró cumplir con los objetivos iniciales, proporcionando recomendaciones basadas en la preferencia de temática del usuario. El clasificador de emociones y el sistema de similitud semántica demostraron ser herramientas efectivas para capturar la intención del usuario.
* La implementación de NER también contribuyó a mejorar la precisión en la búsqueda de coincidencias, aumentando la relevancia de las recomendaciones en función de palabras clave específicas en las frases del usuario.

**2. Limitaciones y Desafíos**

* Uno de los principales desafíos fue la interpretación de frases complejas, ya que el modelo BERT, aunque es preciso en la mayoría de los casos, puede experimentar dificultades en contextos poco comunes.
* El proceso de scraping para el dataset de libros también presentó limitaciones en cuanto a disponibilidad de datos y estandarización del contenido. Sin embargo, se realizaron ajustes para filtrar y limpiar la información, asegurando que los datos fueran utilizables en el sistema de recomendación.
* La duración del scraping y el procesamiento del texto fue otro desafío significativo. Debido al gran volumen de información en los tres datasets (películas, juegos de mesa y libros), el tiempo necesario para obtener y parsear todo el contenido fue considerable. El parsing involucró el manejo de textos extensos en descripciones, lo cual incrementó el tiempo de procesamiento y el consumo de recursos.
* En particular, la longitud de algunos textos requería ajustes adicionales para asegurar que fueran interpretados correctamente por el sistema de recomendación, sin afectar su rendimiento.

**3. Recomendaciones para Trabajos Futuros**

* En el futuro, sería beneficioso ampliar el conjunto de categorías de recomendaciones y realizar un ajuste más específico para reconocer diferentes estilos y géneros dentro de cada categoría.
* La implementación de un modelo de NLP ajustado a varios idiomas podría mejorar aún más la utilidad del sistema, especialmente en la clasificación de emociones y el reconocimiento de entidades.
* Dado el tiempo considerable que requiere el scraping y el parsing de textos largos en los tres datasets (películas, juegos de mesa y libros), sería recomendable encontrar técnicas de optimización para acelerar este proceso.
* De igual forma, mejorar la velocidad de parsep y el rendimiento general del sistema, sin comprometer la calidad de la recomendación. Además, se podría evaluar el impacto de la longitud de los textos en el modelo y ajustar los preprocesamientos para mejorar la eficiencia en futuras implementaciones.

## 

## **Referencias**

Unidad 1 - Extracción y Procesamiento de Texto: <https://gentle-cress-e61.notion.site/Unidad-1-Extracci-n-y-Procesamiento-de-Texto-25f15e55f7ac4e1096376b060df1837f>

Unidad 2 - Representación Vectorial de Texto: <https://gentle-cress-e61.notion.site/Unidad-2-Representaci-n-Vectorial-de-Texto-6ad0dcf3b18a447e8b2989b325b57d3f>

Unidad 3 - Procesamiento del Lenguaje: <https://gentle-cress-e61.notion.site/Unidad-3-Procesamiento-del-Lenguaje-48b3f630e08a49e59bbcabfa39273e0c>

## **Anexos**

**1. Código Fuente**

* El código Python completo, incluyendo el clasificador de emociones, el procesamiento de embeddings, y el scraping para el dataset de libros, se incluye en el archivo adjunto del informe.

**2. Datos Brutos**

* Datasets preprocesados y archivos CSV generados durante el scraping también se adjuntan para su referencia y replicación del proceso.

**Link Google Drive:** <https://drive.google.com/drive/folders/1zcbWzUJZDywuySo-YuFyQbqOrgZaFWDl?usp=sharing>