

Ejercicio: Implementación de un Buscador Flexible en Python

Ejercicio 1:

Objetivo: Desarrollar un buscador flexible en Python que permita encontrar frases relevantes en un conjunto de textos en español, utilizando técnicas de procesamiento de lenguaje natural.

- 1) Desarrollar una función **procesar_texto(texto)** que dado un texto de entrada, permita normalizar, eliminar stopwords, y lematizar.
- 2) Escribir una lista de 10 frases, que utilizaremos para realizar búsquedas.
- 3) Escribir una función que permita, a partir de una frase, realizar una búsqueda sobre las 10 frases, comparando con las frases procesadas con la función del punto 1. En la comparación, calcularemos un puntaje de acuerdo a las palabras en común, sin importar el orden.
- 4) Mostrar un ranking descendente de las frases según el puntaje

Ejercicio 2:

Objetivo: Desarrollar un programa en Python que permita al usuario buscar expresiones en lunfardo, basándose en descripciones de conceptos o acciones. El sistema deberá comparar la descripción del usuario con las definiciones de un diccionario de lunfardo y devolver las tres palabras cuyas definiciones sean más parecidas a la descripción ingresada.

Preparación de Datos:

Extraiga el texto de un diccionario de lunfardo desde un recurso en línea (enlace proporcionado), y organícelo en una tabla de dos columnas:

- Columna 1: Palabra en lunfardo.
- Columna 2: Definición de la palabra.

Desarrollo del Programa:

Implemente una función **`buscar_en_lunfardo(descripcion)`** que realice de las siguientes tareas las que crea útiles:

- Normalización: Convertir la descripción a minúsculas y eliminar caracteres no alfanuméricos.
- Eliminación de stopwords: Remover palabras que no aporten significado relevante, utilizando una lista predefinida de stopwords en español.

- Lematización: Reducir las palabras a su forma base o lema, para facilitar la comparación con las definiciones del diccionario.
- Tokenización: Dividir la descripción en palabras individuales para su análisis.
- Comparación y Conteo: Utilizar técnicas básicas de comparación de texto para identificar las tres definiciones en el diccionario que más se asemejen a la descripción ingresada.

Ejemplo:

Entrada del usuario: "Persona que engaña a los demás."

Salida esperada: Podría incluir palabras como "vivo", "avivado", "pícaro", dependiendo de las definiciones disponibles en el diccionario de lunfardo.

Recursos:

Diccionario de lunfardo ([link](#)).

Ejercicio 3:

Objetivo: Desarrollar un programa en Python que permita al usuario buscar expresiones en lunfardo, basándose en descripciones de conceptos o acciones. El sistema deberá comparar la descripción del usuario con las definiciones de un diccionario de lunfardo y devolver las tres palabras cuyas definiciones sean más parecidas a la descripción ingresada.

Utilice la técnica que crea más conveniente del a la unidad 2 para mejorar la performance de las respuestas. (Sólo utilice metodologías hasta word2vec).

Analice las mejoras de comportamiento del algoritmo de recomendación.