Memoria del Proyecto: Cálculo de sentimiento

# 1. Planteamiento del Problema

El objetivo de este código es procesar un conjunto de tweets y calcular el sentimiento de las palabras que no tienen un valor preasignado utilizando un archivo de sentimientos. Para ello, el programa procesa dos archivos de entrada: uno contiene los sentimientos asociados a palabras o conjuntos de palabras (Sentimientos.txt) y otro contiene los tweets en formato JSON (Tweets.txt). Tras leer y limpiar los tweets, se calculan los sentimientos de las palabras en los tweets y se asignan valores a aquellas que no tienen un sentimiento asociado. Además, el programa permite filtrar palabras ya conocidas y asignarles un valor predeterminado.

# 2. Enfoque y Metodología

## 2.1 Orden de instrucciones

1. **Lectura de Archivos**: El programa comienza leyendo el archivo de sentimientos (Sentimientos.txt) y el archivo de tweets (Tweets.txt). El primero es un archivo de texto donde cada línea contiene una palabra o conjunto de palabras seguida de un valor numérico, que indica su "sentimiento". El segundo es un archivo JSON que contiene el texto de los tweets.
2. **Procesamiento del Archivo de Sentimientos**: En el archivo Sentimientos.txt, cada línea se divide en una palabra y su correspondiente valor de sentimiento, que se almacena en un diccionario. Este diccionario se utilizará posteriormente para calcular el sentimiento de los tweets.
3. **Preprocesamiento de Tweets**: Los tweets pueden contener una gran cantidad de ruido que debe eliminarse antes de analizarlos. Para ello, el código incluye una función llamada limpiar\_tweet() que realiza una serie de operaciones de limpieza en cada tweet.
4. **Cálculo del Sentimiento de los Tweets**: Una vez limpiado el tweet, el código calcula el sentimiento sumando los valores de sentimiento de las palabras que se encuentran en el tweet, utilizando el diccionario de sentimientos previamente cargado. Si alguna palabra no se encuentra en el diccionario, se puede asignar un valor predeterminado, como el valor del tweet completo o el valor promedio del sentimiento.
5. **Asignación de Sentimiento a Palabras Desconocidas**: Si se encuentran palabras en el tweet que no tienen un sentimiento asociado, se les puede asignar un valor del sentimiento total del tweet o un valor calculado como el promedio del sentimiento del tweet, dependiendo de la estrategia elegida.

## 2.2 Función limpiar\_tweet()

La función limpiar\_tweet() realiza una serie de transformaciones utilizando expresiones regulares para eliminar todos los elementos no deseados del texto del tweet. Esto incluye menciones, hashtags, enlaces, emojis, caracteres no latinos, signos de puntuación, fechas, números y códigos HTML. El propósito es garantizar que solo se trabaje con el texto relevante del tweet, lo que mejora la precisión del análisis de sentimientos.

### Descripción Detallada de la Función limpiar\_tweet

La función recibe un tweet como texto y realiza las siguientes tareas de limpieza:

#### Eliminar "RT":

Se elimina la palabra "RT" que aparece en los retweets, ya que no aporta valor al análisis del contenido del tweet.



#### Eliminar Menciones, Hashtags y Enlaces:

Se eliminan las menciones a usuarios (por ejemplo, @usuario), los hashtags (por ejemplo, #hashtag) y los enlaces (por ejemplo, http:// o https://). Estos elementos son irrelevantes para el análisis de sentimientos y pueden generar ruido en el texto.

#### Eliminar Emojis:

Se eliminan todos los emojis del tweet, ya que los emojis no suelen ser interpretables por el sistema de análisis de sentimientos y pueden dificultar el proceso de extracción de datos relevantes.

#### Eliminar Caracteres No Latinos:

Se eliminan caracteres que no son latinos (como caracteres chinos, japoneses, griegos, etc.), ya que estos no son relevantes para el análisis de los tweets escritos en caracteres latinos.

#### Eliminar Signos de Puntuación y Caracteres Especiales:

Se eliminan signos de puntuación como comas, puntos, signos de interrogación, comillas y otros caracteres especiales, que no aportan significado en el análisis del sentimiento del tweet.

#### Eliminar Fechas:

Se eliminan las fechas en formatos comunes como DD/MM/YY, YYYY-MM-DD, entre otros. Las fechas no contribuyen al análisis de sentimientos.

#### Eliminar Números Sueltos:

Se eliminan los números que aparecen en el tweet, ya que, en la mayoría de los casos, no aportan un valor útil para el análisis de sentimientos.



#### Eliminar Códigos HTML:

Se eliminan fragmentos de texto que representan códigos HTML (por ejemplo, &amp;, &lt3;), que pueden aparecer en los tweets por diversas razones y no tienen relevancia para el análisis.

#### Eliminar Espacios Extra y Verificar Vacío:

Se eliminan los espacios en blanco al principio y al final del tweet. Si el tweet resultante está vacío tras la limpieza, se devuelve None.

# 3.Eficiencia y Optimización

En este proyecto, se han implementado diversas estrategias para mejorar la eficiencia y optimización del proceso de análisis de sentimientos de los tweets. A continuación, se detallan las principales técnicas utilizadas:

1. **Uso de Diccionarios para Almacenamiento de Sentimientos**:
   * Se emplea un diccionario en Python (valores) para almacenar las palabras y sus valores de sentimiento. Esta estructura de datos permite realizar búsquedas de forma eficiente, ya que el tiempo de acceso promedio en un diccionario es O(1). Esto significa que la búsqueda de los valores de sentimiento para cada palabra es rápida y eficiente.
2. **Procesamiento Línea por Línea**:
   * El archivo de tweets se procesa línea por línea utilizando un ciclo for. Esto permite manejar archivos grandes sin cargar todo el archivo en memoria al mismo tiempo, lo que optimiza el uso de la memoria, especialmente cuando se trabaja con grandes volúmenes de datos. Así, se reduce el riesgo de que el programa consuma demasiada memoria y se vuelva ineficiente.
3. **Eliminación de Elementos Innecesarios con Expresiones Regulares**:
   * La función limpiar\_tweet() utiliza expresiones regulares para eliminar menciones, hashtags, enlaces, emojis, caracteres no latinos, signos de puntuación, fechas y números del texto de los tweets. Este enfoque basado en expresiones regulares es eficiente y permite limpiar rápidamente el texto sin tener que procesar cada elemento de manera manual, mejorando la eficiencia del preprocesamiento de los tweets.
4. **Manejo de Errores con try-except**:
   * Se emplean bloques try-except para manejar posibles errores, como la lectura de archivos no encontrados o líneas de JSON mal formateadas. Esto permite que el programa continúe su ejecución sin interrumpirse, incluso si se encuentran problemas con algunos de los datos de entrada.
5. **Optimización en la Lectura de Archivos**:
   * Se utilizan las funciones open() y with para la apertura de archivos. La instrucción with asegura que los archivos se cierren automáticamente después de ser leídos, lo que optimiza el uso de los recursos del sistema y evita fugas de memoria. Además, se especifica el uso de encoding="utf-8" para asegurar que se manejen correctamente los caracteres especiales en los archivos.
6. **Reducción del Costo Computacional**:
   * El cálculo del sentimiento de los tweets se realiza de manera eficiente al dividir el texto en palabras y buscar cada palabra en el diccionario de sentimientos. Solo se suman los valores de las palabras que están en el diccionario, y las palabras desconocidas se omiten o se les asigna un valor predeterminado, lo que reduce el tiempo de procesamiento al evitar operaciones innecesarias.

# 4. Resultados Esperados

El objetivo del proyecto es calcular el "sentimiento del tweet" y asignar un valor a las palabras que no tienen un sentimiento previamente asignado. Los resultados esperados son los siguientes:

1. **Cálculo del Sentimiento del Tweet**:
   * Para cada tweet procesado, se debe obtener un valor de "sentimiento" basado en la suma de los valores de las palabras que están en el diccionario de sentimientos. Si una palabra no tiene un valor asociado, se le asigna un valor predeterminado (0 en este caso). El valor total del tweet será la suma de los sentimientos de las palabras presentes en el tweet.
2. **Resultado en Pantalla**:
   * El programa imprimirá por pantalla, para cada tweet procesado, las palabras que no tienen un sentimiento asociado y les asignará un valor. Este valor puede ser el sentimiento total del tweet o el valor promedio del sentimiento, dependiendo de la estrategia utilizada.

**Ejemplo de salida en pantalla**:

'I': 3

'will': 3

'go': 3

'on': 3

'vacation': No mostramos esta palabra, dado que tiene un sentimiento ya asociado anteriormente.

'to': 3

'the': 3

'beach': No mostramos esta palabra, dado que tiene un sentimiento ya asociado anteriormente.

1. **Archivo de Resultados**:
   * Los resultados también se guardarán en un archivo de texto (Resultados\_Sentimientos\_Desconocidos.txt). En este archivo, se registrará cada palabra desconocida junto con el sentimiento asignado. El formato será el mismo que el de salida por pantalla.
2. **Manejo de Tweets Válidos y Vacíos**:
   * Los tweets que no contienen contenido relevante (por ejemplo, aquellos que están vacíos después de la limpieza) no se procesarán y no aparecerán en los resultados. El sistema está diseñado para filtrar los tweets vacíos y solo procesar aquellos que contienen texto significativo.
3. **Resultados con Palabras Desconocidas**:
   * En aquellos casos en los que un tweet contiene palabras desconocidas (es decir, palabras que no están en el diccionario de sentimientos), el sistema asignará un valor predeterminado (como el sentimiento total del tweet o el valor promedio). Estas palabras y su sentimiento asociado serán impresos en pantalla y guardados en el archivo de resultados.

**Ejemplo de asignación de valores a palabras desconocidas**: Si el tweet es:

"I will go on vacation to the beach"

Y las palabras "vacation" y "beach" ya tienen un valor en el archivo de sentimientos, el sistema asignará el valor total del tweet a las demás palabras, como "I", "will", "go", etc.

1. **Informes de Errores**:
   * En caso de que se encuentren errores (como tweets mal formateados o archivos no encontrados), el programa los manejará sin interrumpir su ejecución, imprimiendo mensajes de error para alertar al usuario sobre los problemas detectados.

# 6. Conclusión

El código implementado permite leer y procesar tweets, calculando el sentimiento de las palabras que no tienen un valor preasignado y asignando un valor calculado a las palabras desconocidas. La función limpiar\_tweet() juega un papel crucial en este proceso al eliminar los elementos no relevantes y asegurarse de que el análisis de sentimientos se realice sobre un texto limpio y consistente. Este enfoque mejora la precisión y la fiabilidad del análisis de datos de Twitter.

Laura Rodríguez Corpas