Caso 3: Sentimientos en Twitter

Twitter es una red social que ha degradado a una comunidad llena de Trolls (trollear es la acción de hacer sentir mal o hacer enojar a alguien con bromas pesadas o comentarios fuera de lugar) o Cuñados (aquellos que comentan sobre cualquier asunto, queriendo aparentar ser más listos que los demás). El dicho "No alimentes al Troll" no se aplica en Twitter y estos se retroalimentan hasta el punto de conseguir que conversaciones de alto nivel se conviertan en un patio de escuela. Entre Trolls y Cuñados Twitter ya no es lo que era.

Gracias a las técnicas de análisis de texto (*Text Analysis*) podemos combatir los Trolls y Cuñados y devolver Twitter a sus años de esplendor. *Text Analysis* consiste en extraer información a partir de datos de lenguaje humano para comprender cómo otros seres humanos entienden el mundo, agruparlos y extraer patrones de comportamiento.

En este Caso 3 debes realizar un análisis sentimental, sintáctico y gramatical de comentarios Twitter. La base de datos la puedes descargar desde eStudy (Caso 3 dataset), la cual contiene un CSV de mensajes enviados a Twitter con las siguientes columnas:

1. Puntuación sentimental o polaridad (-5 = negativa … 0 = neutral … 5 = positiva) (por calcular)
2. Id del tweet
3. Fecha del tweet (Sat May 16 23:58:44 UTC 2009)
4. Búsqueda. Si no hay búsqueda, el valor es NO_QUERY
5. Usuario que ha tweeteado
6. Texto del tweet

Con estos datos se os propone que apliquéis técnicas analíticas y de visualización para responder a las siguientes preguntas. No hay restricciones acerca de las técnicas ni tecnologías a utilizar siempre y cuando los resultados sean reproducibles y estén debidamente justificados. No obstante, las siguientes librerías y códigos de ejemplo os pueden ser muy útiles para responderlas:

Librería NLTK
https://www.nltk.org/install.html

Propósito: Trabajar con datos en lenguaje humano.

Librería texstat
https://pypi.org/project/textstat/

Propósito: Calcular estadística a partir de datos en lenguaje humano.

Unsupervised-Text-Clustering using Natural Language Processing (NLP)

Para realizar un conglomerado analítico de un corpus documental/textos se acostumbra a seguir los siguientes pasos genéricos. La técnica consiste en crear un vector cuantitativo a partir de los textos, previa limpieza y transformación, para aplicar técnicas de conglomerado:

1. Eliminar caracteres de puntuación, espacios adicionales, dígitos u otros caracteres que puedan entorpecer el análisis textual
2. Tokenizar y eliminar *Stopwords*. Se requiere un diccionario de palabras para quitar aquellas que puedan entorpecer el análisis textual. Por ejemplo, se puede utilizar "*from nltk.corpus import stopwords*". Ejemplo: NLTK stop words - Python Tutorial (pythonspot.com)
3. Encontrar la raíz de las palabras aplicando *lemmatization* o *stemming*.
4. Aplicar vectorizado del tokenizado para calcular apariciones de los tokens y cuantificar los tweets. Se pueden usar distintos cálculos, por ejemplo Bag-of-Words, Word2Vec, o TFIDF con "*from sklearn.feature_extraction.text import TfidfVectorizer*"
5. Aplicar clustering con técnicas adecuadas. Por ejemplo, Kmeans previo cálculo del número de clusters con técnicas como Elbow.

Transformers

Huggingface pone a disposición una manera muy asequible de realizar análisis sentimentales con modelos pre-entrenados. Sigue estos dos enlaces para poder realizar las preguntas extras del caso:

1. Getting Started with Sentiment Analysis using Python
2. Pipelines

ANÁLISIS:

1. ¿Cuál es la distribución de las polaridades y complejidad de lectura/escritura de los tweets en el dataset?
    a. ¿Hay una mayor cantidad de tweets positivos, negativos o neutrales?
    b. ¿Cómo se relacionan las distintas polaridades según la complejidad de lectura/escritura de los tweets?
2. ¿Existen patrones gramaticales o sintácticos comunes en los tweets con polaridad positiva o negativa? Por ejemplo, puede que los tweets positivos tiendan a utilizar más palabras de agradecimiento o elogios, mientras que los tweets negativos utilizan más palabras de crítica o enojo.
3. ¿Qué usuarios tienden a generar tweets con una polaridad más positiva o negativa? ¿Hay alguna relación entre la polaridad de los tweets y el número de seguidores de un usuario?
4. ¿Hay alguna palabra o conjunto de palabras específicas que estén asociadas con tweets de polaridad extrema?
    a. ¿Estas palabras son más comunes en tweets sobre un tema en particular o están distribuidas en todo el dataset?
    b. Escoge un tema y clusteriza los usuarios según polaridades.
5. ¿Hay alguna correlación entre la polaridad de un tweet y la fecha en que se publicó?
    a. ¿Los tweets publicados durante ciertos períodos de tiempo tienden a ser más positivos o negativos que otros?
6. Identifica los Top 10 Trolls y Top 10 Influencers. Justifica las características de un usuario Troll e Influencer.
7. **Extra:** Utiliza Transformers con el pipeline de Huggingface para calcular la polaridad de los tweets y comparar los resultados de la pregunta 1.

VISUALIZACIÓN:

1. ¿Cómo se distribuyen los tweets según su polaridad a lo largo del tiempo?
2. Visualiza el análisis sintáctico (número de palabras, frase, verbos, nombres…) de los top 10 Trolls e Influencers.
3. ¿Existe alguna correlación entre el número de seguidores de un usuario y la polaridad de sus tweets? Representa visualmente esta relación.
4. Crea una nube de palabras para cada polaridad.

5. ¿Cómo se distribuyen los tweets según su polaridad en función de la hora del día o el día de la semana?