# Práctica 1 Minería de Texto - Minería de Medios Sociales.

Curso: 2018/2019. Néstor Rodríguez Vico. DNI: 75573052C - nrv23@correo.ugr.es

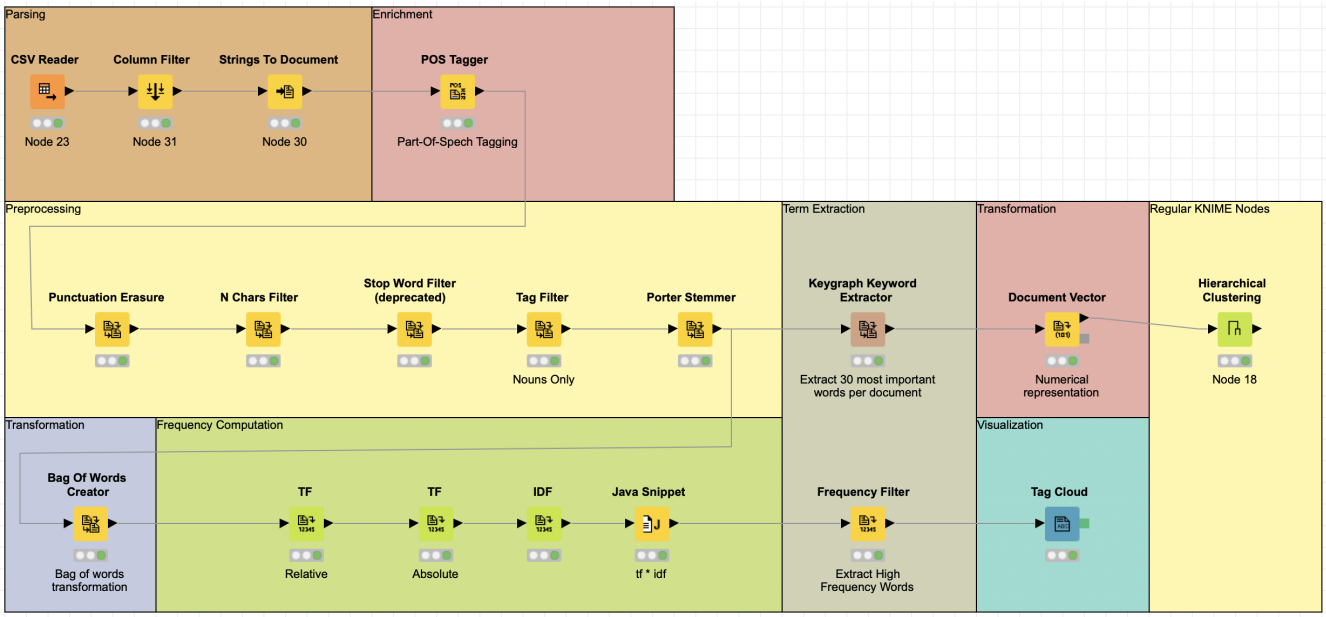
20 de mayo de 2019

## 1. Introducción.

Los datos usados para realizar esta práctica han sido obtenidos tras realizar la búsqueda *Documents classification* en *Google*. Dicha búsqueda lleva a muchas competiciones de *Kaggle*. Muchas de ellas son privadas (se necesita una invitación para poder descargar los dataset) pero en una de ellas (https://www.kaggle.com/yufengdev/bbc-fulltext-and-category) se puede descargar un conjunto de datos proveniente de la página de la BBC (http://mlg.ucd.ie/datasets/bbc.html). Esta colección consiste en *2225* documentos provenientes del sitio web de noticias de la *BBC*, correspondientes a historias de cinco áreas temáticas de 2004-2005. Dichas áreas son: negocios, entretenimiento, política, deporte, tecnología. Dicho conjunto de datos cuenta con *2225* documentos.

## 2. Workflow.

El *workflow* desarrollado es el siguiente:




Se ha cogido el *workflow* proporcionado y se han hecho los siguientes cambios:

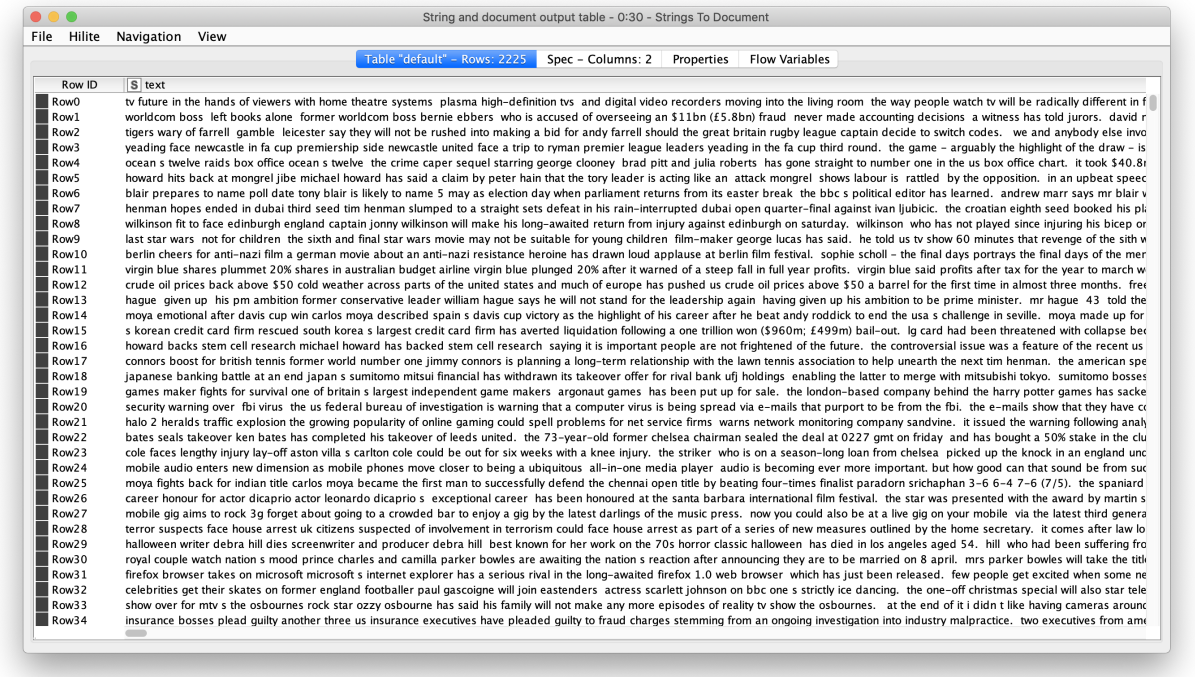
- Se ha cambiado como se obtiene la colección de documentos. En este caso se ha empleado un lector de *CSV*. Una vez leidos los documentos, mandamos la información a un nodo que flitra las columnas para quedarnos solo con la columna que contiene la información del documento en sí.

- Una vez tenemos los datos limpios, convertimos el texto que representa el documento a una variable de tipo *Document*, que es el tipo usado por *KNIME* para representar un documento.
- En la sección de enriquecimiento he quitado el nodo *Abner Tagger* ya que es un preprocesamiento específico para el problema original propuesto y no aplica en el problema presentado en esta memoria.

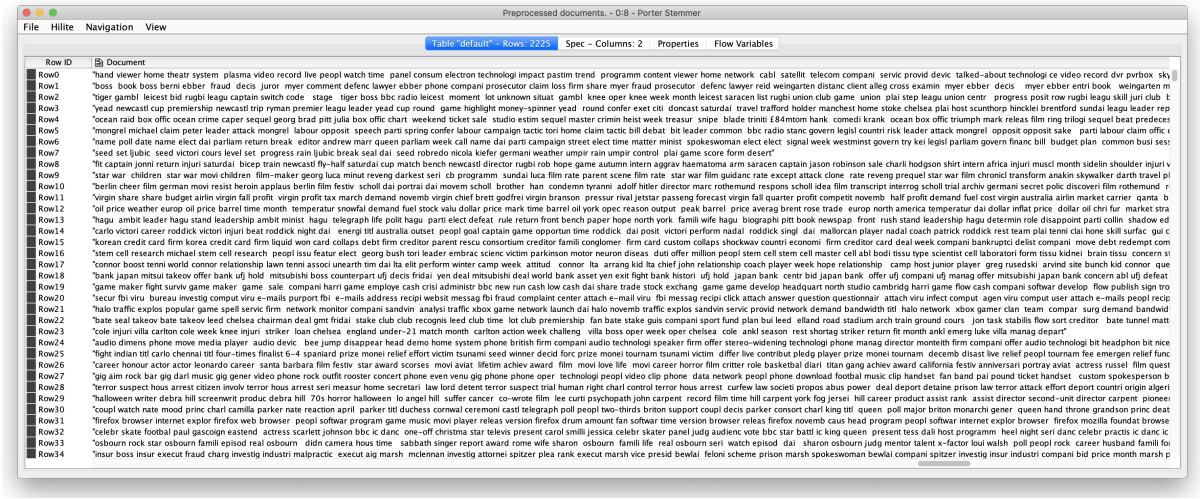
El resto del *workflow* se mantiene de la misma forma:

- Eliminamos los signos de puntuación con el nodo *Punctuation Erasure*.
- Eliminamos las palabas con un tamaño menor que *N*, en nuestro caso 3, con el nodo *N Chars Filter*.
- Eliminamos las palabras vacías con el nodo *Stop Word Filter*. En nuestro caso, hemos usado la lista de *stopwords* proveída por *KNIME* para el idioma del problema, *inglés*.
- Eliminamos las palabras que tienen una etiqueta asociada, es decir, son palabras de una categoría concreta con el nodo *Tag Filter*. Por ejemplo, eliminamos los pronombres, ya que no nos aportan nada.
- Finalmente, aplicamos el algoritmos de *Porter* con el nodo *Porter Stemmer* para quedarnos con la raíz de las palabras.

Tras ejecutar el primer y segundo nodo ya tenemos el texto de los documentos cargados. Para ver el contenido de dicho nodo podemos pinchar en el segundo nodo y elegir la última opción del menú para mostrar los datos, la opción *Filtered table*. El resultado es el siguiente:




Tras ejecutar todos los nodoes de la sección *Parsing*, de la sección *Enrichment* y de la sección *Preprocessing* tenemos un conjunto de datos ya preprocesado y limpio. Podemos ver como ha quedado dicho conjunto si hacemos click derecho en el nodo *Porter Stemmer* y seleccionamos la última opción del menú, la opción *Preprocessed documents*. El resultado es el siguiente:




Si comparamos los resultados podemos ver las diferencias que hemos aplicado en los nodos. Por ejemplo, podemos ver como el documento representado en la línea identificada por *Row3* ha desaparecido el punto que hay. También podemos observar como han desaparecido las palabras con una longitud menor que *3*. Aunque los cambios más notables son cómo hemos sido capaces de eliminar las *stopwords*, eliminar ciertas palabras que tienen una etiqueta asociada y el quedarnos con la raiz de las palabras. Estas tres técnicas nos permiten limpiar los documentos de palabras vacías de contenido (como pueden ser las *stopwords* o las palaras que tienen una etiqueta asociada y no nos interesan, como pueden ser los pronombres). Finalmente, nos hemos quedado con la raiz de las palabras, lo cual permite generalizar que texto es usado para entrenar nuestro modelo. Esta idea permite reconocer la palabra *comer* y la palabra *comes* como la misma, ya que su raiz es la misma.

Finalmente, usamos un nodo *Tag Cloud*, el cual genera una nube de etiquetas. Una nube de etiquetas es una representación de palabras que indica la importancia de las palabras de una forma visual, jugando con el tamaño y la transparencia del texto mostrado. En mi caso, las palabras más relevantes son *tottenham*, *librari* y *godzilla*, tal y como podemos ver en la nube de etiquetas generada por el *workflow*: