Proyecto de Minería de Datos: Borrador de memoria completa

Markus Fischer • Guzmán López • David Pérez • Ander Raso

1 Introducción

1.1 Objetivo de la tarea

En este proyecto se nos encarga la tarea de trabajar en el campo del *Text Mining* y del clustering de documentos. Nuestro objetivo es, a partir de una gran colección de textos, buscar formas de agruparlos y tratar de extraer conclusiones de los resultados que obtengamos.

1.2 Propuesta de trabajo del grupo

Como grupo, hemos decidido realizar la tarea sobre la colección de autopsias verbales que se propuso como opción. Esto se debe a que, ademas de considerar el tema muy interesante, creemos que una propuesta que se encuentra muy próxima al uso que se da al *Text Mining* en el ámbito científico.

Sin embargo, de esta decisión también surgen ciertos retos a los que debemos poner solución:

- Muchos de los reportes de autopsia están en un lenguaje poco preciso y muchas veces ininteligible, probablemente causado tanto por el desconocimiento de los que dieron el reporte, como por las traducciones que se han hecho a estos.
- En muchas ocasiones no hay reporte verbal o este es irrelevante, por lo que la única información útil de que se dispone es de los datos del fallecido, tales como país, edad, sexo, etc.
- TODO: Añadir alguna más para que no quede vacío.

2 Descripción y análisis de datos

3 Preproceso de datos

El formato original del archivo de autopsias es un xlsx, es decir, una hoja de cálculo de Microsoft Office. Esto nos facilita mucho la tarea de procesar los textos, ya que es muy fácil convertir de xlsx a formatos de texto plano. En nuestro caso elegimos csv, ya que es un formato muy limpio y que más adelante nos será útil para transformarlo al formato arff, el cual es compatible con Weka

y gran parte de las librerias de Data Mining que hemos utilizado.

Una vez tenemos nuestro documento como csv, procedemos a preprocesarlo. Ya que los valores identificador, grupo de edad, lugar, diagnóstico, sexo y edad están codificados segun unos criterios que se mantienen a lo largo de todas las instancias, sabemos que no debemos preprocesarlos. Sin embargo, es en el campo de la respuesta verbal en el que tenemos que solucionar algunas anomalías:

- **Mayúsculas y minúsculas:** Para evitar diferencias entre palabras escritas completamente en mínuscula, completamente en mayúscula o con la primera letra mayúscula, debemos transformar todas ellas a un mismo formato. En nuestro caso el formato escogido son las minúsculas.
- **Saltos de línea:** En algunos casos nos encontramos con saltos de línea internos en el texto. Ya que csv separa sus diferentes atributos por comas y las instancias por líneas, los saltos de línea internos en el texto crean inconsistencias y errores al procesar el csv. El método más fácil para desacerse de ellos pero mantener el texto en el mismo formato es sustituírlos por espacios.
- **Símbolos:** Aquí tenemos varios problemas que debemos arreglar de diferentes formas.
  - **Barras "/":** en ocasiones nos encontramos con dos palabras escritas de modo "palabra1/palabra2". Para evitar que se procesen ambas como una misma palabra, realizamos el mismo procedimiento que utilizamos con los saltos de línea: cambiamos las barras por espacios.
  - **Corchetes "[ ]":** en ocasiones nos encontramos con dos palabras escritas de modo "palabra1/palabra2". Para evitar que se procesen ambas como una misma palabra, realizamos el mismo procedimiento que utilizamos con los saltos de línea: cambiamos las barras por espacios.

4 Clustering

5 Evaluación

6 Experimentos

7 Conclusiones

8 Bbliografía