



## Trabajo Práctico Tratamiento y Análisis del Texto

### Bibliografía

[MIR] Capítulo 7, [TOL] Capítulo 3, [MAN] Capítulos 6 (parcial)

- 1) Escriba un programa que realice operaciones simples de análisis léxico sobre una colección (documentos de texto en un directorio) y calcule algunas medidas básicas sobre la misma. Debe establecer criterios para (revise la bibliografía):
  - ¿Cómo definir una "palabra" (o término)?
  - ¿Cómo tratar números y signos de puntuación?
  - Tratar direcciones de correo electrónico para que se extraigan correctamente.

Como salida, el programa debe generar:

- a) Un archivo (terminos.txt) con la lista de términos a indexar (ordenado), su frecuencia en la colección y su DF.
- b) Un segundo archivo (estadisticas.txt) con los siguientes datos:
  - Cantidad de documentos procesados
  - Cantidad de *tokens* y términos extraídos
  - Promedio de *tokens* y términos de un documento
  - Largo promedio de un término
  - Cantidad de *tokens* y términos del documento más corto y del más largo
  - Cantidad de términos que aparecen sólo 1 vez en la colección
- c) Un tercer archivo con:
  - La lista de los 10 términos más frecuentes (y su TF)
  - La lista de los 10 términos menos frecuentes (y su TF)

Explique para qué utilizaría la información extraída.

- 2) El programa del ejercicio anterior, ¿Funciona de la misma manera si la colección está en Inglés? Justifique y ejemplifique. En caso que la respuesta sea negativa, realice una nueva versión con las modificaciones necesarias.
- 3) A partir de su programa del ejercicio 1, incluya un proceso de Stemming. Puede implementar un stemmer clásico como Porter (para el inglés) a partir de una librería Perl. Si procesa la colección en español puede probar con: <http://snowball.tartarus.org/algorithms/spanish/stemmer.html> ó con la versión que se encuentra en <http://stemmer-es.sourceforge.net>

Luego de modificar su programa, corra nuevamente el proceso del ejercicio 1 y analice los cambios en la colección. ¿Qué implica este resultado? Busque ejemplos de pares de términos que tienen la misma raíz pero que el stemmer los trató diferente y términos que son diferentes y se los trató igual.



- 4) Nuevamente, modifique su programa del ejercicio 1 para que elimine las palabras vacías (contenidas en un archivo de texto). Ejecute el proceso del ejercicio 1 y analice los cambios. ¿Qué ocurrió? ¿Qué conclusión se obtiene?
- 5) En este ejercicio se propone verificar la predicción de ley de Zipf. Para ello, descargue desde Project Gutenberg el texto del Quijote de Cervantes (<http://www.gutenberg.org/dirs/etext99/2donq10.zip>) y escriba un programa que extraiga los términos y calcule las frecuencias (no debe trabajar demasiado porque ya lo hizo para el ejercicio 1). Con dichos datos y los estimados por Zipf grafique ambas distribuciones (haga 2 gráficos, uno en escala normal y otro en log-log). ¿Cómo se comporta la predicción? ¿Qué conclusiones puede obtener? Repita el análisis podando un porcentaje  $x$  de los términos más y menos frecuentes. ¿Con qué porcentaje de poda se mejora la predicción para este texto?
- 6) De acuerdo a la ley de Heaps, ¿Qué proporción de una colección debe ser procesada para encontrar el 90% del vocabulario? (Suponga  $\beta = 0.5$ )
- 7) Existe un requerimiento consistente en indexar un corpus cuyo tamaño es de 800.000 palabras. Del análisis de pequeñas muestras del corpus se obtuvo que para 10.000 palabras, el tamaño del vocabulario es de aproximadamente 25.000 términos únicos y para 50.000 palabras de aproximadamente 78.000. ¿Cómo puede estimar el tamaño final del vocabulario del corpus completo? ¿Qué cantidad de memoria se requiere para almacenarlo? (va a necesitar información extra). ¿Qué sucedería en caso que el corpus tenga 1.000.000.000 de palabras?
- 8) Suponiendo que las frecuencias de los términos en una colección siguen la ley de Zipf con parámetro 0.1 ¿Cuál es el número menor de términos comunes que juntos suman el 25% de las ocurrencias?