CAIM Lab, Session 3:
User Relevance Feedback

David Carballo Montalbán

David Sànchez Peña

Implementación

Para implementar el script de *Rocchio*, hemos seguido el guión proporcionado en el enunciado de la práctica.
Para facilitarnos el trabajo en el momento de hacer los cálculos, decidimos pasar todas las variables como parámetros cuando llamamos el script. Para representar las queries, hemos optado por utilizar diccionarios, donde guardamos los términos de la query y sus respectivos pesos. En la primera iteración, los pesos de todos los términos valen 1.
Antes de aplicar la regla de *Rocchio*, calculamos el *TFIDF* de los *k* documentos más importantes utilizando el script que implementamos en la anterior sesión.

Experimentos

Para los experimentos utilizaremos la colección de documentos de *News*.

- *nrounds*:
  Después de realizar varios experimentos en los que cambiamos el valor de *nrounds* y mantenemos el valor de las otras variables (a=2,b=1,k=5,R=10), podemos ver que la relevancia de los documentos va aumentando a cada iteración. También vemos que en distintas iteraciones se pueden incorporar nuevos documentos a los resultados. Esto es debido a que cada iteración, la query se distancia más de la original, permitiendo así que se puedan considerar otros documentos relevantes.

- *k*:
  Aumentar la *k* lo único que hace es aumentar el numero de documentos que mostramos en el resultado. Por lo tanto, la llamada al script k=10 nos devuelve los mismos documentos que devuelve el script con k=5, además de 5 documentos con menor puntuación que los 5 primeros. Esto puede llevar a mostrar documentos poco relevantes y a aumentar el número de falsos positivos, reduciendo así la *precisión* y aumentando el *recall*.

- R:
  El parámetro R nos sirve para controlar cuántos términos tendremos en la nueva query. De esta manera, obtendremos más o menos precisión en los resultados respecto al valor escogido para R.

- *a* y *β*:
  Tal y como se puede ver en la fórmula de *Rocchio*, aumentar el valor de *a* hace que demos prioridad a la query proporcionada por el usuario. Por el contrario, aumentar el valor de *β*, da prioridad a la query formada por los *TFIDF*. Esto lo podemos ver experimentalmente, donde independientemente del valor de *β*, si vamos aumentando el valor de *a*, también lo hace el valor de los pesos de las queries resultantes. Si en distintas llamadas al script asignamos a *a* los valores 3, 4 y 5, veremos que los valores de los pesos irán aumentando en potencias de 2, 3 y 4 respectivamente.

Conclusiones

Después de observar como afecta cada parámetro a los resultados obtenidos, podemos llegar a ciertas conclusiones. Entre ellas, podemos ver que las variables que más influyen en los resultados son *a*, *β* y nrounds; debido a que éstos son los parámetros que regulan la puntuación de cada elemento de las queries.
Por el otro lado tenemos las variables *k* y *R*, que nos pueden servir para ajustar la *precisión* y el *recall*, dependiendo del tipo de respuesta que queramos dar.