1 Objetivo

Desarrollar una aplicación para evaluar la precisión y la exhaustividad (precision & recall) de un motor de búsqueda.

2 Descripción

El formato debe ser exactamente el siguiente:

```
pr score.tsv judgment.tsv
```

donde `pr` es la aplicación que toma dos ficheros, `score.tsv` y `judgment.tsv` en entrada y retorna los resultados `stdout`.

2.1 Formatos

La aplicación puede ser escrita en cualquier lenguaje que compile o interprete en la máquina `dagobert.selfip.org`. Candidatos son C, C++, java, python, perl, etc.

Cada linea de `score.tsv` se compone de 3 campos separados por una tabulación cada uno. El primer campo es el identificador de una consulta, el secundo campo es el identificador de un documento y el tercer campo es el grado de relevancia del documento a la consulta (un float de 0. a 1.). Si un par consulta / documento no aparece en este fichero, el grado de relevancia asociado es cero por defecto.

Las lineas de `judgment` están compuestas por dos campos separados por una tabulación. El primer campo contiene el identificador de una consulta y el secundo campo el identificador de un documento. Se supone que los pares consulta / documento que aparecen en este fichero son todos relevantes, y que los que no aparecen son todos irrelevantes.

En salida, el programa `pr` produce lineas que empiezan con el identificador de la consulta seguido de los 101 grados de precisión que corresponden a exhaustividades (recall) de 0 a 100% por pasos de 1%. Los campos están separados por tabulaciones.

3 Condiciones de entrega

La tarea debe ser entregada antes del jueves 25 de octubre a las 23h59 de acuerdo a los pasos que se describen aquí. Es la única manera de entregar la tarea. En particular, si la aplicación de entrega mencionada abajo retorna un mensaje de error, la tarea no esta considerada como entregada.

La máquina donde hacer la entrega es un Debian Linux 2.6.17 con un AMD Athlon(tm) XP 2600+ e 1Gb de RAM con dirección `dagobert.selfip.org`. Es accesible desde `dichato` y `anakena`.

Pare acceder a la maquina se debe usar `ssh`. Para obtener una cuenta en esta máquina, enviar un mail a `bpiwowar@dcc.uchile.cl` con título CC52D login=XXX donde XXX es reemplazado por su login en la red del DCC.

El código debe estar adecuadamente comentado, no explicando qué hace cada línea de código, pero sí cómo funciona y qué algoritmo usan. La estructura de directorios a utilizar es la siguiente:

README.txt	Su nombre y la descripción de lo que contiene cada archivo de código fuente.
src	Código fuente de los programas que entrega
bin	Programa `pr`
doc	Documentos que quiere que leamos sobre su tarea, opcional
etc	Todo lo demás, este directorio no será revisado, opcional

Para entregar, ejecute (en `dagobert.selfip.org`): /home/cceval/2007/cc52d/pr/entrega AAAAAA— donde AAAAAA es el nombre del archivo (con un formato zip, tar.gz, etc.) que tiene la estructura descrito más arriba. El comando comprueba:

- que la estructura del archivo este correcta.
- que la aplicación `pr` funciona correctamente.

Si estas dos condiciones están cumplidas, la tarea esta entregada. En el caso contrario, un mensaje de error aparece. No hay límites sobre el número de entregas, y una nueva entrega borra la anterior.

Especial cuidado debe tenerse con los ficheros de datos `score.tsv` y `judgment.tsv` ya que Linux y Windows usan codificación distintos para los saltos de linea. Se recomienda tentar compilar y ejecutar su programa en su cuenta en `dagobert.selfip.org` para verificar que todo funciona bien.