

Descrição

O trabalho consiste na implementação de um programa que gera um índice invertido para uma base de documentos no contexto de um sistema de Recuperação da Informação (RI).

A base de documentos

A base de documentos é composta por um conjunto arbitrário de arquivos de texto. Assuma que nesses arquivos texto, palavras são separadas por um ou mais dos seguintes caracteres: espaço em branco (), ponto (.), vírgula (,), exclamação (!) ou interrogação (?). Assuma também que todo o conteúdo dos arquivos na base está em caracteres minúsculos e sem caracteres acentuados.

A entrada do programa

Seu programa deverá receber dois argumentos como entrada **pela linha de comando**. O primeiro argumento especifica o caminho de um arquivo texto que contém os caminhos de todos os arquivos que compõe a base, cada um em uma linha. O segundo argumento especifica o caminho de um arquivo texto que traz uma lista com *stopwords*, com uma *stopword* por linha. As *stopwords* são palavras que **não** possuem significado semântico para um sistema de RI, e portanto, **não devem ser consideradas na construção do índice invertido**.

Exemplo: Vamos supor que nossa base é composta pelos arquivos a.txt, b.txt e c.txt. Vamos supor também que nosso programa se chama indice.exe. Assim, chamaríamos nosso programa pela linha de comando fazendo:

```
> indice.exe base.txt stopwords.txt
```

onde o arquivo base.txt contém os caminhos para os arquivos que compõe a base de documentos, conforme a seguir:

```
a.txt  
b.txt  
c.txt
```

base.txt

, e o arquivo stopwords.txt possui uma lista de palavras que deverão ser tratados pelo sistema como *stopwords*, isto é, não deverão ser consideradas na construção do índice:

```
o  
de  
na  
em  
nao
```

stopwords.txt

o arquivo deverá gerar um arquivo de saída chamado indice.txt, que conterá o índice invertido gerado a partir dos documentos da base. Para a geração do índice invertido, é preciso considerar cada palavra **não stopword** que apareça em algum dos documentos da base. Para cada uma dessas palavras no índice, é preciso apontar o número do arquivo em que a mesma aparece, e a quantidade de vezes em que a mesma aparece no arquivo. Os arquivos são numerados segundo a ordem em que aparecem no arquivo que indica os documentos da base, que, para o nosso exemplo, foi denominado como base.txt. Assim, o arquivo a.txt é o arquivo 1, o arquivo b.txt é o arquivo 2 e, por fim, o arquivo c.txt é o arquivo 3. Suponha que estes arquivos estejam preenchidos conforme abaixo:

era uma casa muito
engracada. nao tinha teto,
nao tinha nada.

a.txt

quem casa quer casa.
quem nao mora em casa,
tambem quer casa!

b.txt

quer casar comigo, amor?
quer casar comigo,
faca o favor! mora na
minha casa!

c.txt

Nesse, caso, o arquivo indice.txt com o índice gerado seria composto por:

amor: 3,1
casa: 1,1 2,4 3,1
casar: 3,2
comigo: 3,2
engracada: 1,1
era: 1,1
faca: 3,1
favor: 3,1
minha: 3,1
mora: 2,1 3,1
muito: 1,1
nada: 1,1
quem: 2,2
quer: 2,2 3,2
tambem: 2,1
teto: 1,1
tinha: 1,2

indice.txt

Observe que, para cada palavra no índice invertido, temos uma lista de pares a,q onde a é o número do arquivo em que a palavra aparece, e q é a quantidade de vezes em que a palavra aparece no arquivo. Assim, para a palavra casa, por exemplo, temos o par 1,1, indicando que no arquivo 1, este termo aparece uma vez. Em seguida, temos o par 2,4, indicando que no arquivo 2 este termo apareceu 4 vezes. Por fim, temos o par 3,1, indicando que, no arquivo 3, este termo aparece uma vez. **Note que as stopwords não devem entrar no índice invertido!** Não é estritamente necessário que o índice

invertido seja gerado em ordem alfabética. Todavia, gerar o índice em ordem poupará esforço na próxima etapa do trabalho.

Linguagens permitidas

O programa que gera o índice invertido pode ser desenvolvido em qualquer uma das seguintes linguagens:

- C
- C++
- Java
- Python
- Haskell

Entrega

O código fonte pode ser desenvolvido em grupo de até 3 pessoas, e a entrega por e-mail ao professor: wendelmelo@ufu.br.