Universidade Federal do Rio de Janeiro (UFRJ)
Departamento de Ciência da Computação (DCC)

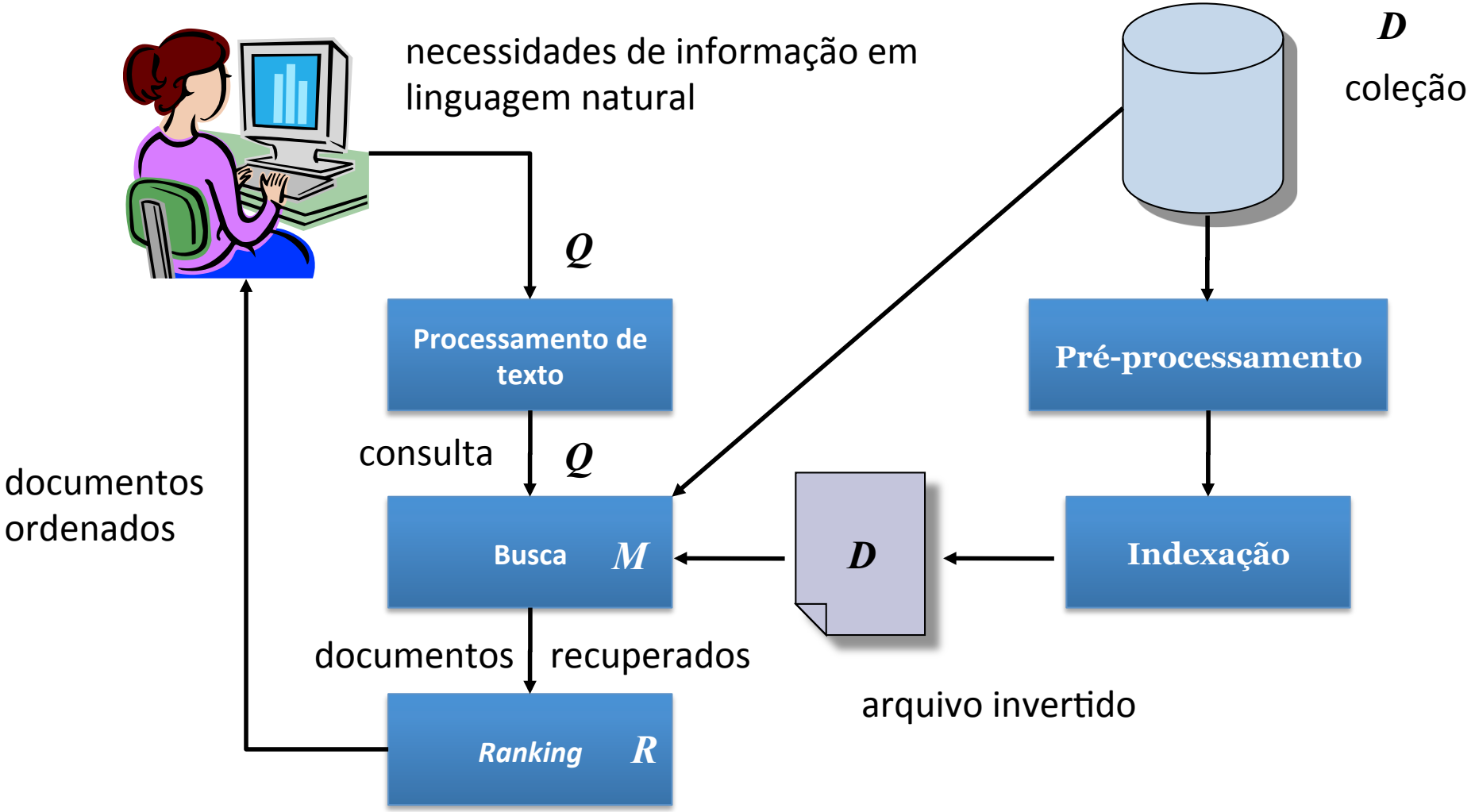
DEPARTAMENTO DE
CIÊNCIA DA
COMPUTAÇÃO
- UFRJ -

Recuperação da Informação (MAB605)

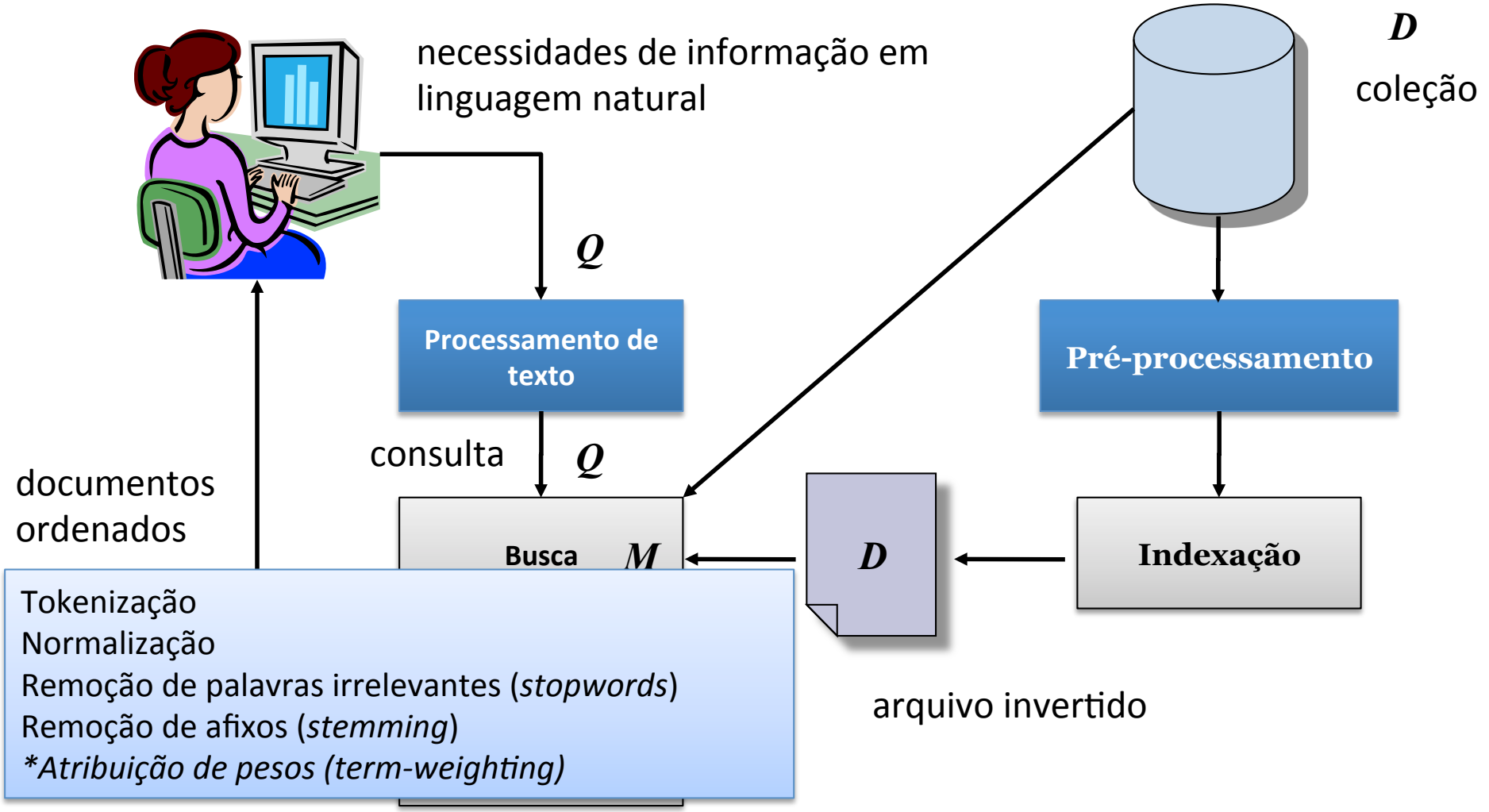
Pré-processamento

Profa. Giseli Rabello Lopes

Componentes de um Sistemas de RI Relembrando...



Componentes de um Sistemas de RI Relembrando...



Roteiro

- Introdução
- Pré-processamento de documentos
  - Análise léxica do texto
  - Eliminação de *stopwords*
  - *Stemming* (continuação...)
  - Seleção de palavras-chave
  - Tesauros
- Referências

Pré-processamento de documentos
[Baeza-Yates & Ribeiro-Neto, 2013]

- Pode ser dividido em cinco operações (ou transformações) textuais:
  1. Análise léxica do texto
  2. Eliminação de *stopwords*
  3. S*temming* das palavras remanescentes
  4. Seleção de termos de índice ou palavras-chave
  5. Construção de estruturas de categorização de termos (tesauros) - detalhes em aula posterior sobre expansão de consulta

Stemming (Relembrando...)

[Baeza-Yates & Ribeiro-Neto, 2013]

- Usuário especifica palavra em uma consulta mas apenas uma variante dessa está presente em um documento relevante
  - Plurais, gerúndios e sufixos que indicam passado (variações sintáticas)
  - Parcialmente resolvido pela adoção de *stems*
- ***Stem***
  - Porção de uma palavra que resta após a remoção de afixos (prefixos/sufixos)
  - Ex.: **connect** – stem de *connected, connecting, connection, connections*

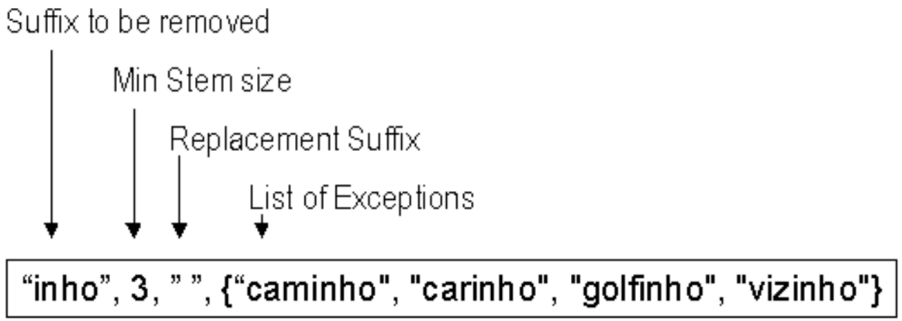
Stemming

[Baeza-Yates & Ribeiro-Neto, 2013]

- Tipos de estratégias de *stemming*:
  - Busca em tabela: busca do *stem* de uma palavra em uma tabela
  - Remoção de afixos: na qual a parte mais importante é a remoção de sufixos
  - Variedade de sucessores: determina os limites dos morfemas e utiliza conhecimento da linguística estrutural
  - N-gramas: identifica digramas e trigramas (*clustering* de termos)

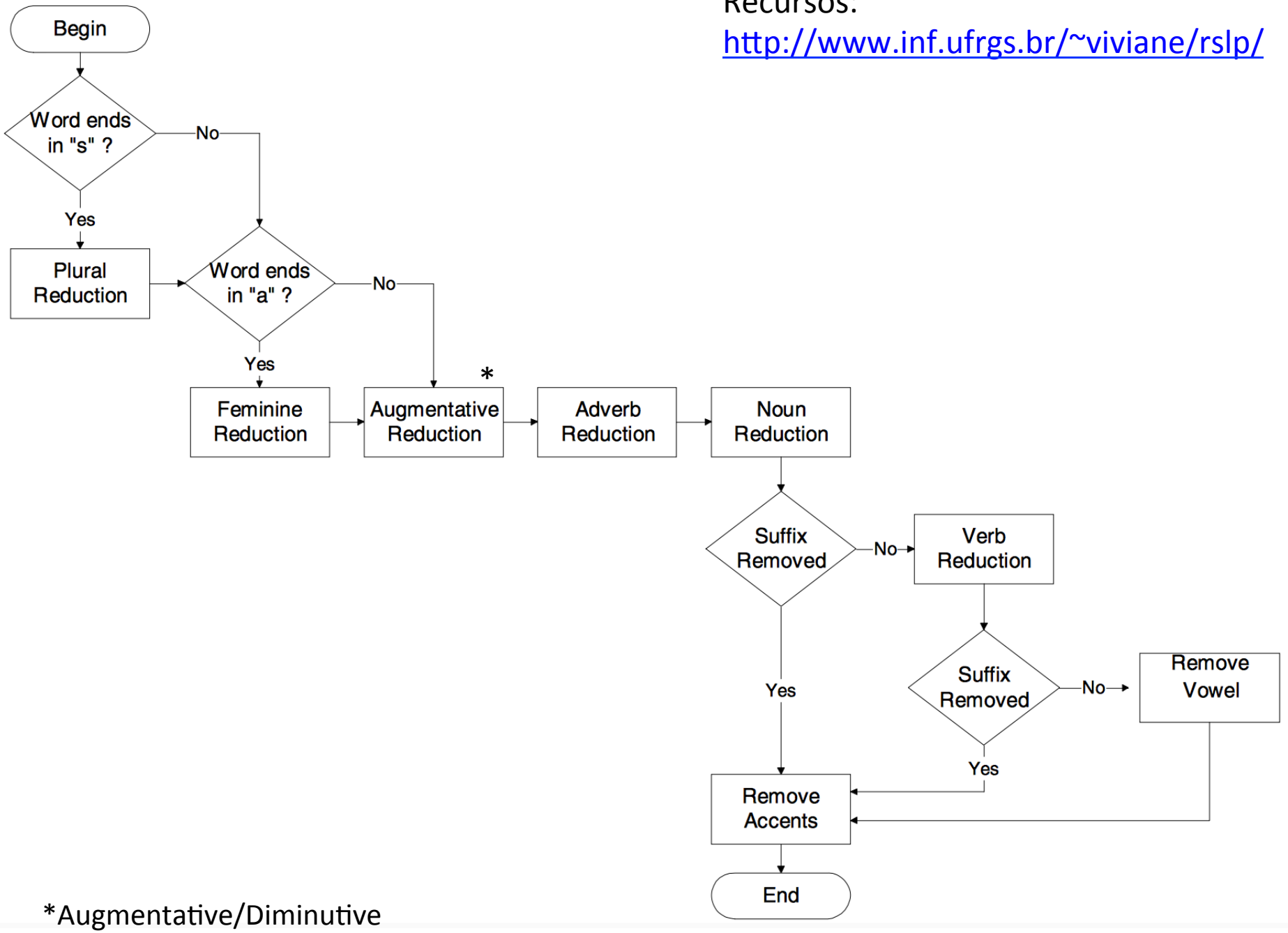
Stemming – RSLP

- **RSLP** (Removedor de Sufixos da Língua Portuguesa)
  - V.M. Orengo & C. Huyck. A Stemming Algorithm for the Portuguese Language. SPIRE, 2001.
  - Baseado em um conjunto de regras



  - Regras agrupadas em 8 passos
  - Em cada passo apenas uma regra é aplicada

Recursos:
http://www.inf.ufrgs.br/~viviane/rslp/

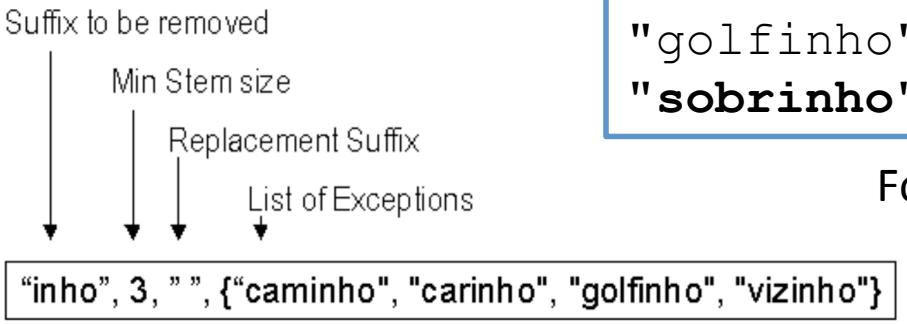


*Augmentative/Diminutive

Exercício

- Crie 3 regras de *stemming* para redução de plural no formato:

```
"inho", 3, "", {"caminho",
"carinho", "cominho",
"golfinho", "padrinho",
"sobrinho", "vizinho"}
```



Fonte: [Orengo et al., 2007]

- Após, busque regras definidas no RSLP que sejam similares às criadas por você e estabeleça um comparativo.

Exercício

- # Step 1: Plural Reduction
  { "Plural", 3, 1, {"s"},
  {"ns",1,"m"},
  {"ões",3,"ão"},
  {"ães",1,"ão",{"mãe"}},
  {"ais",1,"al",{"cais","mais"}},
  {"éis",2,"el"},
  {"eis",2,"el"},
  {"óis",2,"ol"},
  {"is",2,"il",{"lápis","cais","mais","crúcis","biquínis","pois","depois","dois","leis"}},
  {"les",3,"l"},
  {"res",3,"r"},
  {"s",2,"",{"aliás","pires","lápis","cais","mais","mas","menos",
  "férias","fezes","pêsames","crúcis","gás",
  "atrás","moisés","através","convés","ês",
  "país","após","ambas","ambos","messias"}}}

Stemming – RSLP

- Dificuldades:
  - Exceções: para quase todas as regras existem exceções
  - Homógrafos: ex. casais (casal ou casar?)
  - Verbos: redução para o infinitivo; verbos irregulares
  - Mudanças no radical: emitir - emissão
  - Nomes próprios

Stemming – Variedade de sucessores

- A variedade de sucessores de uma *string* é o número de caracteres diferentes que a sucedem em um *corpus*
- Quanto mais longa for a *substring*, menor será seu número de sucessores
- Etapas do processo:
  1. Determinar a variedade de sucessores para uma palavra
  2. Segmentar a palavra
  3. Escolher um dos segmentos como *stem*

Stemming – Variedade de sucessores

1. Determinar a variedade de sucessores
   - Corpus: able, ape, beatable, fixable, read, readable, reading, reads, red, ripe, rope
   - Palavra teste: **readable**

Prefixo	Variedade de sucessores	Letras
r	3	e, i, o
re	2	a, d
rea	1	d
read	3	a, i, s
reada	1	b
readab	1	l
readabl	1	e
readable	1	-

Fonte: [Frakes & Baeza-Yates, 1992]

Stemming – Variedade de sucessores

2. Métodos de segmentação
   - Cutoff
     - Um valor de limiar é determinado
     - Quando o valor é alcançado, um limite entre *strings* é identificado
     - Problema: como selecionar o valor de cutoff?
       - Se for muito pequeno, cortes incorretos serão feitos
       - Se for muito grande, cortes corretos não serão identificados

Stemming – Variedade de sucessores

2. Métodos de segmentação

– Peak and plateau

- Um limite é colocado sempre que o número de sucessores de um caracter for maior do que o seu antecessor imediato
- Vantagem: elimina o problema de se escolher um limiar

Prefixo	Variedade de sucessores	Letras
r	3	e, i, o
re	2	a, d
rea	1	d
read	3	a, i, s
reada	1	b
readab	1	l
readabl	1	e
readable	1	-

Fonte: [Frakes & Baeza-Yates, 1992]

Stemming – Variedade de sucessores

2. Métodos de segmentação
   - Complete word
     - Um limite é definido após uma *substring* se ela for uma palavra completa no corpus

Corpus: able, ape, beatable, fixable, read, readable, reading, reads, red, rope, ripe

Palavra teste: read | able

Stemming – Variedade de sucessores

3. Seleção do *stem*
   - Depois de segmentar a *string*, devemos escolher qual dos segmentos é o *stem*
   - Hafer & Weiss usaram a seguinte regra:
     - Se o primeiro segmento ocorre em 12 ou menos palavras do *corpus*, então o primeiro segmento é o *stem*. Caso contrário, o segundo segmento é o *stem*.
     - Mesmo que a palavra tenha mais do que 2 segmentos, o *stem* será selecionado somente entre o 1º e o 2º.

Exercício

- Simular as etapas do processo de *stemming* por variedade de sucessores para a palavra invariável
  (Obs.: usar o método de segmentação *Peak and plateau*)

  – Corpus: adequado, coerente, idioma, ilegal, inadequado, inato, incoerente, incrível, indeciso, indevido, inexistente, infeliz, injusto, invariante, invariável, invencível, invisível, justo, varia, variabilidade, variável, visível

Stemming – N-gramas

- Um ***n*-grama** é um conjunto de ***n*** caracteres consecutivos
- Ex.: digramas (*n*=2)

  banana → ba an na an na (5 digramas, 3 únicos)

  bananada → ba an na an na ad da (7 digramas, 5 únicos)
- Nesse método, nenhum *stem* é produzido
- Calcula-se o coeficiente Dice

$$S = 2C / (A + B)$$

  - Onde:
    - A é o número de n-gramas únicos na primeira palavra
    - B é o número de n-gramas únicos na segunda palavra
    - C é o número de digramas únicos compartilhados

- No ex.: S=(2*3)/(3+5)=6/8=0.75

Stemming – N-gramas

- A similaridade entre os termos é determinada para todos os pares de termos da coleção, formando uma matriz de similaridade
- A matriz resultante é esparsa pois a similaridade para a maioria dos pares é zero
- Aplica-se um algoritmo de *clustering* sobre a matriz usando um limiar para a similaridade
- Estudos em coleções pequenas mostraram bons resultados

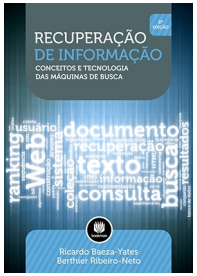
Exercício

- Considere o par de termos:
  - invariável, invariante
- Considerando *n*=3 (trigramas), calcule o coeficiente de Dice entre esses termos.

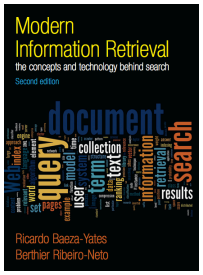
$$S = {2C}/{(A+B)}$$

  - Onde:
    - A é o número de n-gramas únicos na primeira palavra
    - B é o número de n-gramas únicos na segunda palavra
    - C é o número de digramas únicos compartilhados

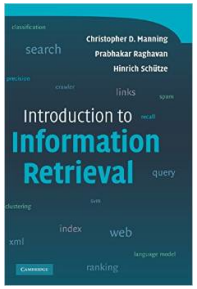
Referências



- Baeza-Yates, R.; Ribeiro-Neto, B. Recuperação de Informação: Conceitos e Tecnologia das Máquinas de Busca. 2 ed. Bookman, 2013.



- Baeza-Yates, R.; Ribeiro-Neto, B. Modern Information Retrieval. Wokingham, UK: Addison-Wesley, 2 ed., 2011.



- Manning, C. D.; Raghavan, P.; Schütze, H. Introduction to Information Retrieval. Cambridge University Press, 2008.

  Online edition 2009: http://nlp.stanford.edu/IR-book/

Referências

- Frakes, W.B.; Baeza-Yates, R. Information Retrieval – Data Structures and Algorithms. Cap. 8, p. 135, 1992, London, UK: Prentice Hall.
- Orengo, V.M, L. Buriol, and A. Coelho. A study on the use of Stemming for Monolingual Ad-Hoc Portuguese Information Retrieval , in Evaluation of Multilingual and Multi-modal Information Retrieval, C. Peters, et al., Editors. 2007, Springer Berlin / Heidelberg. p. 91-98. CLEF 2006, Alicante.

Universidade Federal do Rio de Janeiro (UFRJ)
Departamento de Ciência da Computação (DCC)

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO - UFRJ -

Recuperação da Informação (MAB605)

Dúvidas?

Profa. Giseli Rabello Lopes
giseli@dcc.ufrj.br
CCMN - DCC - Sala E-2012