

Pesquisa e Publicação de Informação

Indexação e Procura (cont.)

Nuno D. Mendes

Licenciatura em Sistemas e Tecnologias de Informação

18 Mai 2012
ISEGI – UNL

Motivação

- ▶ A procura de um termo no texto revela-se necessária quando
 - ▶ Não há um índice construído na colecção de documentos
 - ▶ Existe um índice, mas a sua granularidade não permite determinar a posição exacta das ocorrências
 - ▶ Existe um índice, mas pretende-se fazer a procura de um padrão (termo com erros, ou expressão regular)

Motivação

- ▶ A procura de um termo no texto revela-se necessária quando
 - ▶ Não há um índice construído na colecção de documentos
 - ▶ Existe um índice, mas a sua granularidade não permite determinar a posição exacta das ocorrências
 - ▶ Existe um índice, mas pretende-se fazer a procura de um padrão (termo com erros, ou expressão regular)

Tipos de procura

- 1 Procura sequencial
 - ▶ Força bruta
 - ▶ Algoritmo Knuth-Morris-Pratt
 - ▶ Algoritmo Boyer-Moore

Motivação

- ▶ A procura de um termo no texto revela-se necessária quando
 - ▶ Não há um índice construído na colecção de documentos
 - ▶ Existe um índice, mas a sua granularidade não permite determinar a posição exacta das ocorrências
 - ▶ Existe um índice, mas pretende-se fazer a procura de um padrão (termo com erros, ou expressão regular)

Tipos de procura

- 1 Procura sequencial
 - ▶ Força bruta
 - ▶ Algoritmo Knuth-Morris-Pratt
 - ▶ Algoritmo Boyer-Moore
- 2 Procura aproximada (emparelhamento de padrões)
 - ▶ Procura com erros
 - ▶ Procura de expressões regulares

Definição

O problema da procura de um padrão P num texto T pode ser formalizado da seguinte forma:

Seja Σ um alfabeto de todos os caracteres que ocorrem em T . Procurar o padrão P no texto $T \in \Sigma^*$ consiste em determinar todas as ocorrências de P em T . Se a procura for exacta ou aproximada, então $P \in \Sigma^*$, se se tratar da procura de uma expressão regular, então o padrão P exprime um conjunto de elementos de Σ^* , i.e., $P \in 2^{\Sigma^*}$, segundo uma expressão regular.

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P = abracadabra
 T = abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P = abracadabra
 T = abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P = abracadabra
 T = abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abra c abracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abr a cabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abracabracadabra

Força Bruta

Definição

É o algoritmo mais simples de procura. Consiste em testar o padrão P para todas as posições do texto T . Dado que existem $O(n)$ posições no texto e que em cada posição é preciso testar a ocorrência de um padrão de tamanho $O(m)$, este algoritmo tem a complexidade $O(nm)$, no pior caso.

Exemplo

P	=	abracadabra
T	=	abracabracadabra

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	00.....
P	=	abracadabra
T	=	abracabracadabra

A posição j da tabela N indica o tamanho do maior prefixo de $P[0..j]$ que também é sufixo.

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000.....
P	=	<u>a</u> b racadabra
T	=	abracabracadabra

A posição j da tabela N indica o tamanho do maior prefixo de $P[0..j]$ que também é sufixo.

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	0000.....
P	=	<u>a</u> b <u>r</u> acadabra
T	=	abracabracadabra

A posição j da tabela N indica o tamanho do maior prefixo de $P[0..j]$ que também é sufixo.

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	00001.....
P	=	<u>a</u> br̄ac a dabra
T	=	abracabracadabra

A posição j da tabela N indica o tamanho do maior prefixo de $P[0..j]$ que também é sufixo.

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010.....
P	=	<u>a</u> bra c adabra
T	=	abracabracadabra

A posição j da tabela N indica o tamanho do maior prefixo de $P[0..j]$ que também é sufixo.

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	0000101.....
P	=	<u>a</u> brac <u>a</u> dabra
T	=	abracabracadabra

A posição j da tabela N indica o tamanho do maior prefixo de $P[0..j]$ que também é sufixo.

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	00001010....
P	=	<u>a</u> braca <u>d</u> abra
T	=	abracabracadabra

A posição j da tabela N indica o tamanho do maior prefixo de $P[0..j]$ que também é sufixo.

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010101...
P	=	<u>a</u> bracad ā bra
T	=	abracabracadabra

A posição j da tabela N indica o tamanho do maior prefixo de $P[0..j]$ que também é sufixo.

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	0000101012..
P	=	<u>a</u> bracad <u>a</u> bra
T	=	abracabracadabra

A posição j da tabela N indica o tamanho do maior prefixo de $P[0..j]$ que também é sufixo.

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	0000101012 <u>3</u> .
P	=	<u>a</u> bracad <u>a</u> bra
T	=	abracabracadabra

A posição j da tabela N indica o tamanho do maior prefixo de $P[0..j]$ que também é sufixo.

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010101234
P	=	<u>abracad</u> abra
T	=	abracabracadabra

A posição j da tabela N indica o tamanho do maior prefixo de $P[0..j]$ que também é sufixo.

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010101234
P	=	abracadabra
T	=	abracabracadabra

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010101234
P	=	abracadabra
T	=	abracabracadabra

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010101234
P	=	abracadabra
T	=	abracabracadabra

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010101234
P	=	abracadabra
T	=	abracabracadabra

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010101234
P	=	abracadabra
T	=	abracabracadabra

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010101234
P	=	abracadabra
T	=	abracabracadabra

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010101234
P	=	abracadabra
T	=	abracabracadabra

O emparelhamento falha na posição 7 depois de exactamente 7 caracteres emparelhados. Podemos avançar $7 - N[7] = 6$ posições. Tendo começado na posição 0, avançamos para a posição 6. (No caso da falha ocorrer na posição 0, avançamos sempre uma posição).

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010101234
P	=	abracadabra
T	=	abracabracadabra

Knuth-Morris-Pratt

Definição

Evita a comparação de posição do texto com o padrão, usando informação sobre tentativas anteriores. Depende da construção de uma tabela N , baseada na estrutura do padrão P . O algoritmo permite uma complexidade $O(n)$.

Exemplo

N	=	000010101234
P	=	abracadabra
T	=	abracabracadabra

Cada posição do texto é comparada, no máximo, 2 vezes com o padrão, logo temos uma complexidade $O(2n) = O(n)$.

Procura

Procura sequencial

Boyes-Moore

Definição

Família de algoritmos que utiliza uma tabela semelhante a N , mas considera uma janela do tamanho do padrão e realiza o emparelhamento de trás para a frente. Toma o máximo de duas regras heurísticas^a para decidir o melhor deslocamento no texto. 1) a **regra do bom sufixo**, que indica a penúltima ocorrência no padrão do sufixo já emparelhado; 2) a **regra do mau caracter**, que indica a primeira posição à esquerda do sufixo já emparelhado que corresponde ao caracter que causou a falha. Em média tem uma complexidade menor do que o algoritmo de Knut-Morris-Pratt (porque nem todas as posições são comparadas), mas no pior caso tem a complexidade da abordagem de força bruta.

^asub-aproximações ao deslocamento óptimo que não perdem ocorrências do padrão

Ilustração

P	=	abraca d abra
T	=	<u>abraca</u> b <u>abra</u> cadabra
Bom sufixo		abraca <u>b</u> <u>abrac</u> adabra
Mau caracter		abraca <u>b</u> abracadabra

Procura

Procura aproximada

Procura com Programação Dinâmica

Definição

Procura o padrão P no texto T a menos de e erros (inserções/deleções e não-emparelhamento), usando uma recorrência para preencher uma tabela, C .

$$\begin{aligned} C[0, j] &= 0, \quad \forall j \\ C[i, 0] &= i, \quad \forall i \\ C[i, j] &= \begin{cases} C[i-1, j-1] & \text{se } P[i] = T[j] \\ 1 + \min(C[i-1, j], C[i, j-1], C[i-1, j-1]) & \text{se } P[i] \neq T[j] \end{cases} \end{aligned}$$

Ilustração

Procura de survey em surgery com 2 erros

	0	s	u	r	g	e	r	y
0	0	0	0	0	0	0	0	0
s	1	0	1	1	1	1	1	1
u	2	1	0	1	2	2	2	2
r	3	2	1	0	1	2	2	3
v	4	3	2	1	1	2	3	3
e	5	4	3	2	2	1	2	3
y	6	5	4	3	3	2	2	2

Expressão Regular

Uma expressão regular denota um conjunto de sequências de símbolos de um alfabeto Σ e é definida indutivamente da seguinte forma. α é uma expressão regular sse:

- ▶ $\alpha = a$, e $a \in \Sigma$
- ▶ $\alpha = \beta\gamma$, e β, γ são expressões regulares (β concatenado com γ)
- ▶ $\alpha = \beta^*$, e β é uma expressão regular (0 ou mais ocorrências de β)
- ▶ $\alpha = (\beta)$, e β é uma expressão regular
- ▶ $\alpha = \beta|\gamma$ e β, γ são expressões regulares (β ou γ)

Autómato

Autómato determinista para a expressão regular definida sobre $\Sigma = \{a, b\}$: $bb^*(b|b^*a)$

