



Listas invertidas (*Inverted indexes*)


Prof. Dieisson Martinelli

dieisson.martinelli@udesc.br

Programa

- Introdução
- Conceitos básicos
- Listas invertidas completas
- Construção, indexação e busca
- Atividades

Introdução

-  Em ciência da computação, **Lista invertida** (do inglês *inverted list* ou *inverted index*) é uma estrutura de dados que **mapeia palavras-chave** ao seu **conteúdo** ou documento relacionado
- É uma **estratégia de indexação** que permite a realização de **buscas precisas e rápidas**
 - É uma das mais populares estratégias para mecanismos de **obtenção ou recuperação de dados** (*information retrieval*), usada em larga escala em sistemas de gerenciamento de bancos de dados e em serviços de busca (como o Google)

Introdução

- Uma **lista invertida** geralmente é construída com base em uma **lista tradicional** de documentos, e é assim chamada por inverter a **hierarquia da informação**
- Ao invés de uma lista de documentos contendo termos, é obtida uma **lista de termos**, que referenciam estes **documentos**. Esta referência é feita, normalmente, através de um **identificador único**, como uma chave primária
- Junto deste identificador, podem ser armazenadas outras informações, conforme adequado para a natureza das buscas
 - Por exemplo, armazenar a quantidade de vezes que um termo aparece no documento

Introdução



Funcionamento:

- Dada a seguinte lista de documentos...

```
1: "Sei que sou"  
2: "Sou o que sei"  
3: "Sou especial"
```

- Obtém-se a seguinte lista invertida

```
"sei": [1, 2]  
"que": [1, 2]  
"sou": [1, 2, 3]  
"o"   : [2]  
"especial": [3]
```

← Aparece na lista 1 e 2

← Aparece na lista 1 e 2

← Aparece na lista 1, 2 e 3

← Aparece na lista 2

← Aparece na lista 3

Introdução



Aplicações:

- Listas invertidas são um **elemento central** de **sistemas de busca**, pois estes visam trazer **resultados** de **forma rápida e eficiente**
- Buscas por termos em uma **lista tradicional** exige que se **percorra cada documento** e **cada palavra** dentro destes em busca do termo desejado
- Por outro lado, com o uso de uma **lista invertida** pode-se saltar diretamente para o **termo procurado**
- O desempenho tende a ser cada vez mais significativo conforme aumenta o espaço de busca (quantidade de documentos)

Introdução

- O uso de **listas invertidas** tem o potencial de **tornar as buscas mais eficientes**, possibilitando o armazenamento de **informações adicionais** que, acompanhadas de **algoritmos adequados**, facilitam a **classificação** e a **ordenação** dos resultados
- **Desvantagem**: o custo destes benefícios vem na **forma de trabalho adicional** para a manutenção da lista
 - É preciso **manter a lista invertida atualizada** (ou seja, rodar o programa gerador da lista) conforme documentos são inseridos, alterados e excluídos da lista tradicional

Conceitos básicos

- ▶ **Lista invertida** pode ser vista como um mecanismo orientado a palavras utilizado para indexar uma coleção de documentos (geralmente documentos de texto), com o objetivo de acelerar a tarefa de pesquisa
- A estrutura de uma lista invertida é composta por dois elementos: o **vocabulário** e as **ocorrências**
 - **Vocabulário** se refere ao conjunto de **todas as palavras** diferentes do documento texto (palavras **sem repetição**)
 - Para cada palavra do vocabulário há um índice em uma lista que armazena os documentos que contêm aquela palavra

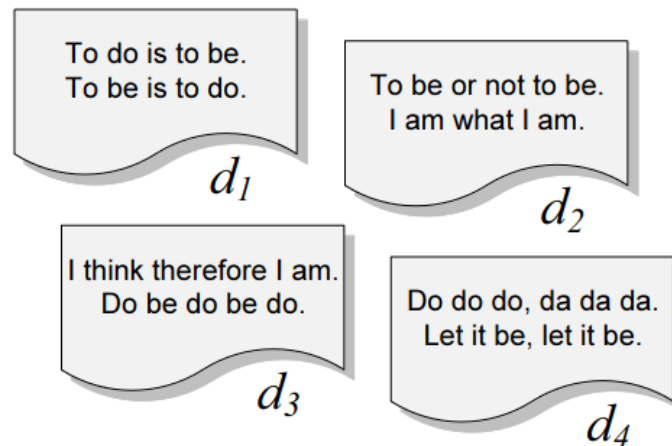
Conceitos básicos

- A maneira mais simples de representar os documentos que contêm cada palavra do vocabulário é com uma **matriz termo-documento**

Vocabulary	n_i	d_1	d_2	d_3	d_4
to	2	4	2	-	-
do	3	2	-	3	3
is	1	2	-	-	-
be	4	2	2	2	2
or	1	-	1	-	-
not	1	-	1	-	-
I	2	-	2	2	-
am	2	-	2	1	-
what	1	-	1	-	-
think	1	-	-	1	-
therefore	1	-	-	1	-
da	1	-	-	-	3
let	1	-	-	-	2
it	1	-	-	-	2

$n_i \rightarrow$ número de documentos onde a palavra aparece

$d_i \rightarrow$ quantidade de vezes que a palavra aparece no documento

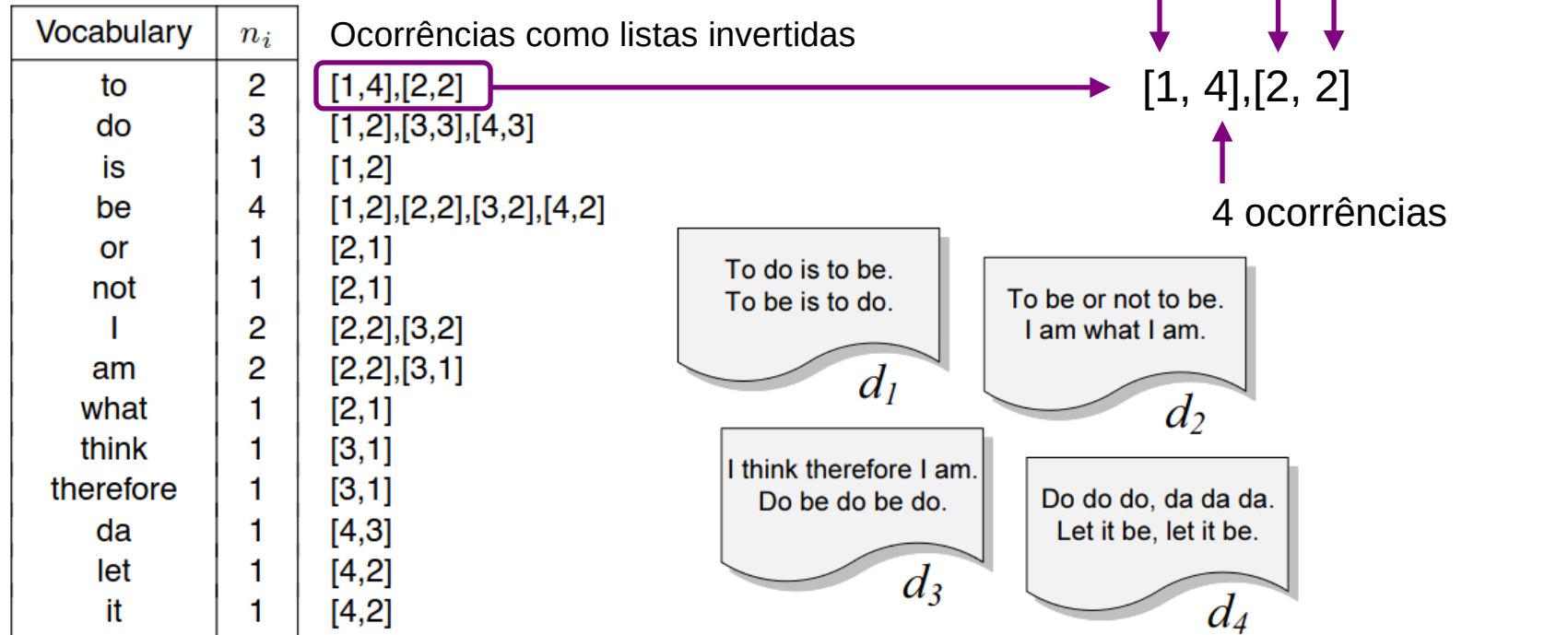


Conceitos básicos

- O principal problema da solução **matriz termo-documento** é que ela **requer muito espaço**
- Possui uma grande quantidade de elementos com valor zero (ou não presentes, desnecessários)
- Como se trata de uma **matriz esparsa**, a solução é associar uma **lista de índices de documentos** a cada palavra
- O conjunto de todas essas listas é chamado de **ocorrências**

Conceitos básicos

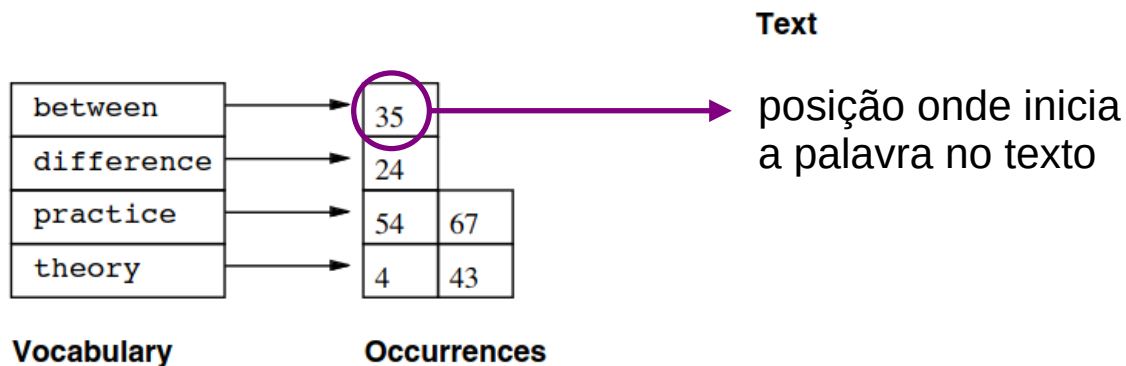
► Exemplo de uma lista invertida básica...



Listas invertidas completas

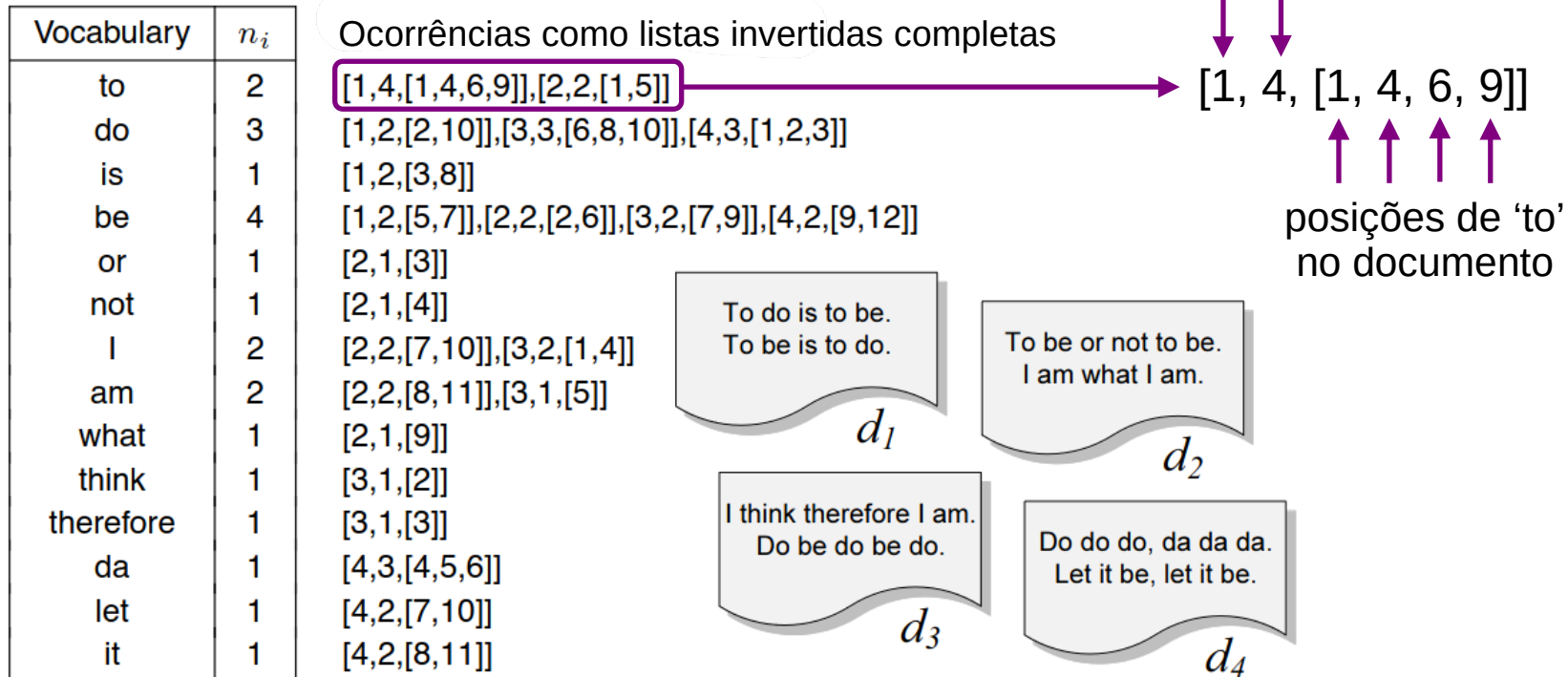
- A lista invertida básica **não é adequada** para responder a **consultas** de palavras em frases ou a **localização** no texto
- Portanto, é preciso adicionar as posições de cada palavra em cada documento, formando uma **lista invertida completa**

1 4 12 18 21 24 35 43 50 54 64 67 77 83
In theory, there is no difference between theory and practice. In practice, there is.



Listas invertidas completas

► No caso de **múltiplos documentos**, é necessário armazenar uma lista de ocorrências por par termo-documento



Exemplo de lista invertida: construção

- Uma **lista invertida** pode ser construída com base em uma **lista comum**, como neste exemplo:

```
Lista_Invertida = {'palavra_1':[(arquivo,posição)],  
                  'palavra_2':[(arquivo,posição)],  
                  'palavra_3':[(arquivo,posição)],  
                  ...  
                  'palavra_n':[(arquivo,posição)]}
```

- Onde:
 - **palavra** é um termo dentro de um arquivo (documento texto)
 - **arquivo** é um número de identificação do arquivo, dentre uma lista de 1 ou mais arquivos, que representam o espaço de busca
 - **posição** é a posição da palavra ou termo no arquivo

Exemplo de lista invertida: indexação

- Considerando três arquivos como **espaço de busca** para o **processo de indexação** de um lista invertida, tem-se:



```
index = {'centro':[(1,1)], 'de':[(1,2),(2,4),(3,2),(3,4)],  
        'educacao':[(1,3)], 'do':[(1,4),(2,2)], 'planalto':[(1,5)],  
        'norte':[(1,6)], 'universidade':[(2,1)], 'estado':[(2,3)],  
        'santa':[(2,5)], 'catarina':[(2,6)], 'disciplina':[(3,1)],  
        'estrutura':[(3,3)], 'dados':[(3,5)], '2':[(3,6)]}
```

- Obs.: os pares (arquivo, posição) poderiam ser tratados externamente, por tuplas em uma lista encadeada para palavras ou termos iguais

Exemplo de lista invertida: busca

- O mecanismo de busca consiste em **obter a localização de um ou mais termos** (palavras) no espaço de busca

```
index = {'centro':[(1,1)], 'de':[(1,2),(2,4),(3,2),(3,4)],  
        'educacao':[(1,3)], 'do':[(1,4),(2,2)], 'planalto':[(1,5)],  
        'norte':[(1,6)], 'universidade':[(2,1)], 'estado':[(2,3)],  
        'santa':[(2,5)], 'catarina':[(2,6)], 'disciplina':[(3,1)],  
        'estrutura':[(3,3)], 'dados':[(3,5)], '2':[(3,6)]}
```

- Se os termos de busca forem "**santa,norte,de,disciplina,udesc**" no espaço de busca da lista invertida acima (p/ arquivos 1.txt, 2.txt e 3.txt), o resultado seria:

```
santa: 2.txt; norte: 1.txt; de: 1.txt, 2.txt 3.txt;  
disciplina: 3.txt; udesc: não encontrado.
```


Atividades

- Desenvolver um código de lista invertida básica para Python e enviar via Moodle