Introdução Algorítmo Complexidade

#### Denilson Alves Pereira Eduardo Emanuel Braga da Silva

Departamento de Ciência da Computação Universidade Federal de Lavras

July 23, 2013

## Objetivo

- Classificar nomes de Veículos de Publicações.
- Ser capaz de reconhecer uma grande variedade de nomes de veículos de publicação.
- Criar um base de dados para treinamento, recolhendo variações dos nomes em bibliotecas digitais.
- Dado uma variação do nome de um veículo, fornece se as informações completas deste veículo.

## Motivação

- Um mesmo veículo tratado como sendo diferente.
- Dificuldades para fazer pesquisas.
- Resultados redundantes nas buscas.
- Inconsistência de dados.

### Dificuldades

Nome	Tipo
Advances in Software Engineering	J
Advances in Engineering Software	J

Table: Dificuldades encontradas.

### Dificuldades

Nome	Tipo
Design Automation Conference	С
PROC DES AUTOM CONF	С

Table: Dificuldades encontradas.

#### Coleta dos dados

- ACM digital library
  - http://dl.acm.org/dl.cfm
- IEEExplore digital library
  - http://ieeexplore.ieee.org/Xplore/home.jsp
- DBLP computer science library
  - http://www.informatik.uni-trier.de/~ley/db/
- wikipedia
  - http://en.wikipedia.org/wiki/List\_of\_compu ter\_science\_conferences
- Qualis
  - http://qualis.capes.gov.br/webqualis/Index.faces

#### Modelo XML

• 
$$< id > 2 < /id >$$

$$\bullet \ < \mathit{acronym} > \mathit{FCRC} < /\mathit{acronym} >$$

$$ullet$$
 < title > FederatedComputingResearchConference < /title >

$$\bullet \ < pub-type > C < /pub-type > \\$$

#### Modelo XML

- < pub venue >
  - $\bullet$  < id > 27 < /id >
  - < idClass > 2027 < /idClass >
  - < acronym > DCFS < /acronym >
  - < title > InternationalWorkshoponDescriptionalComplexityofFormalSystems < /title >
  - ullet < title > WorkshoponDescriptionalComplexityofFormalSystems < /title >
  - < merge of title >
     DescriptionalComplexityofAutomata, GrammarsandRelatedStructures <
     /merge of title >
  - ${\color{red} \bullet} &< merge-of-title > Formal Descriptions and Software Reliability < \\ &/merge-of-title >$
  - ullet < merge of acronym > DCAGRS < /merge of acronym >
  - ullet < merge of acronym > FDSR < /merge of acronym >
  - $\bullet$  < pub type > W < /pub type >
- < /pub venue >

### Estrutura do Arquivo Invertido

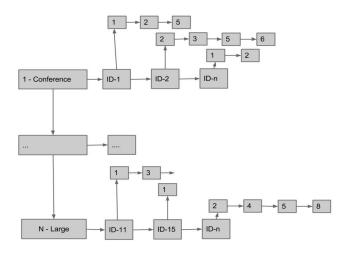


Figure: Estrutura do arquivo invertido.

# Intersecção

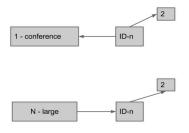


Figure: Interseção entre large e conference.

#### Normalização

- Substituição dos delimitadores " $\backslash/()\{\}[]$ " por espaço em branco.
- Substituição de palavras por outras equivalentes. syposium conference.
- Tranformação de letras em maiúsculo para minúsculo.
- Remoção de espaços em excesso.
- Remoção de qualquer outra pontuação.
- Remoção de stopWords.

#### Algoritmo

```
//Para criar a lista invertida
Para cada veículo nos veículos de treino; faça
| Para cada título, sigla e combinação do título e sigla; faça
| Normalize e crie os tokens.
| Para cada token nos tokens; faça
| | Insira no arquivo invertido o token e sua localização.
//Para criar as regras
Para cada token na lista invertida; faça
| Se o token aparecer em apenas um veículo
| crie a regra com esse token
| Se não
| adicione o token em outra lista de itens.
```

# Algoritmo

Enquanto a lista de itens não for vazia
Para cada item1 nos itens
Para cada item2 nos itens
se apenas a primeira palavra no item1 e no item2 for diferente; faça
Obtenha a lista de intersecção entre as plavras diferentes
Para as outras palavras do iten1; faça
Enquanto a lista não for vazia; faça
faca a lista igual a interseccao desta lista com a lista da palavra.
Se a lista tiver tamanha igual a um
Concatene lexicograficamente a primeira palavra do item2 com o item1
e verifique se existe uma subregra para este resultado.
Se nao existir crie uma nova regra.
Se o tamanho for maior que um
Concatene lexicograficamente o iten1 e o iten2 e adicione em novosItens
Faca itens igual a novos itens

# Algoritmo

//Para fazer o teste
Para cada título a ser classificado; faça
Normalize o título.
Verifique se existe regra para os itensets deste título.
Se existir
Verifique se a similaridade entre o veiculo que aponta a regra e o título é maior que
um threshold
Se foi encontrada pelo menos uma regra então
Obtenha a com maior similaridade e classifique o título.
Se nenhuma regra foi econtrada
Compare o título com toda a base e classifique pela major similaridade

# Complexidade

# Resultados