Documentos

ISSN 1677-9274
Dezembro, 2010

104

Tutorial TaxTools

Embrapa

ISSN 1677-9274
Novembro, 2010

Empresa Brasileira de Pesquisa Agropecuária
Embrapa Informática Agropecuária
Ministério da Agricultura, Pecuária e Abastecimento

Documentos 104

Tutorial TaxTools

Claudia Juliana Poker Moretti
Bruno Malveira Peixoto
Maria Fernanda Moura

Embrapa Informática Agropecuária
Campinas, SP
2010

Embrapa Informática Agropecuária
Av. André Tosello, 209 - Barão Geraldo
Caixa Postal 6041 - 13083-886 - Campinas, SP
Fone: (19) 3211-5700 - Fax: (19) 3211-5754
www.cnptia.embrapa.br
sac@cnptia.embrapa.br

1ª edição on-line 2010

Dados Internacionais de Catalogação na Publicação (CIP)
Embrapa Informática Agropecuária

Moretti, Cláudia Juliana Poker.

Tutorial TaxTools / Claudia Juliana Poker Moretti, Bruno Malveira Peixoto, Maria Fernanda Moura. - Campinas : Embrapa Informática Agropecuária, 2010.

29 p. : il. ; (Documentos / Embrapa Informática Agropecuária, ISSN 1677-9274 ; 104).

1. Mineração de textos. 2. Agrupamento de documentos. 3. Cortes de agrupamento. 5. Visualização de agrupamentos. 6. descrição de agrupamentos. Peixoto, Bruno Malveira. II. Moura, Maria Fernanda. I. Título. II. Série.

CDD 006.33 (21. ed.)

Autores

Claudia Juliana Poker Moretti

Estagiária da Embrapa Informática Agropecuária
Av. André Tosello, 209, Barão Geraldo
Caixa Postal 6041 - 13083-970 - Campinas, SP
e-mail: claupoker@gmail.com

Bruno Malveira Peixoto

Estagiário da Embrapa Informática Agropecuária
e-mail: brunompeixoto@gmail.com

Maria Fernanda Moura

Doutora em ciências de computação e matemática computacional
Pesquisadora da Embrapa Informática Agropecuária
Telefone: (19) 3211-5780
e-mail: fernanda@cnptia.embrapa.br

Apresentação

Com o avanço e a facilidade de acesso à tecnologia da informação, cada vez mais diversas organizações dos mais variados setores têm utilizado sistemas digitais de armazenamento de dados. De acordo com estimativas realizadas em 2006, a quantidade de dados no universo digital atingia 161 hexabytes, o que correspondia a três milhões de vezes a quantidade de informação contida em todos os livros já escritos. Em 2007, a quantidade de informação criada em um ano, superou, pela primeira vez, a quantidade de armazenamento que se pode lidar; e, em 2008, 478 bilhões de gigabytes foram adicionados ao universo digital. Desses dados, 85% do volume é gerado por empresas e 95% dos dados não são estruturados, na sua maioria são imagens e textos. Para que se possa imaginar, essa quantidade de dados, se o universo digital fosse convertido em páginas de texto, poder-se-ia chegar a Plutão e voltar à Terra aproximadamente dez vezes ou cobrir a superfície da Terra como um pacote mais ou menos oito vezes. Nesse ritmo de crescimento, o universo digital deve dobrar a cada dezoito meses (GANTZ; REINSEL, 2010).

Ante a essa grande quantidade de dados, a capacidade humana de analisá-los, compreendê-los e utilizá-los adequadamente é excedida. Dessa forma, tem-se a necessidade de organizar e recuperar a informação, representar conhecimento, identificar tendências etc, com foco nas necessidades específicas de cada empresa. Para transformar e analisar os dados textuais, na Embrapa tem-se lançado mão da ajuda de alguns processos de mineração de textos, melhorando e expandindo o seu uso e exploração. Para isso, tem sido incorporado o uso de algumas técnicas e ferramentas aos processos do dia-a-dia, de modo a melhorá-los sem sobrecarregá-los; e, especialmente de um modo que cada técnica ou ferramenta possa ser configurada para satisfazer a um processo específico.

Algumas alternativas de ferramentas disponíveis no mercado trazem consigo custos de consultoria especializada ou de plataformas de software e hardware a serem adquiridas, sempre em um pacote fechado e com poucas

opções de configuração. Esses pacotes, muitas vezes, acabam por onerar e aumentar a complexidade dos processos habituais. A alternativa que tem-se buscado é desenvolver uma bancada de ferramentas totalmente configuráveis, de fácil integração com novas ferramentas. Dessa forma, pode-se permitir a resolução de problemas de forma barata, configurável e utilizando softwares de domínio público, desenvolvendo ou adaptando ferramentas para as necessidades, e, principalmente, conseguindo inovar em técnicas de mineração de dados. Dentre essas, a TaxTools, um conjunto de técnicas e ferramentas de apoio à construção de taxonomias de tópicos, desenvolvida por laboratório da Universidade de São Paulo, tem sido mantida e evoluída em laboratório da Embrapa Informática Agropecuária. Assim, o presente tutorial procura orientar sua utilização por parte de pessoal técnico previamente treinado em construção de taxonomias de tópicos ou em processos básicos de mineração de textos.

Kleber Xavier Sampaio de Souza
Chefe-Geral
Embrapa Informática Agropecuária

Agradecimentos

Agradecemos ao Ricardo Marcondes Marcacini, aluno de pós-graduação do ICMC ,USP, tanto pela implementação da versão original da TaxTools, quanto pela explicação de seu uso e constantes ajudas.

E, agradecemos ao Labic / ICMC, USP, São Carlos, SP pela cessão da versão original da TaxTools.

Sumário

Tutorial TaxTools

Claudia Juliana Poker Moretti
Bruno Malveira Peixoto
Maria Fernanda Moura

1 Introdução

A ferramenta TaxTools foi desenvolvida pelo Laboratório de Inteligência Computacional (Labic) do Instituto de Ciência Matemática e de Computação (ICMC) da Universidade de São Paulo (USP), campus de São Carlos, SP, com o objetivo de auxiliar no processo de mineração de textos. Atualmente, ela tem sido mantida e evoluída pelo Laboratório de Inteligência Computacional (LabIC) da Embrapa Informática Agropecuária. Esse tutorial abrange apenas as opções disponíveis na TaxTools, que completam o processo de obtenção de uma taxonomia de tópicos (MOURA et al., 2008); como clusterização, cálculos de medidas *intercluster* e de *joinability*, métodos de podas, métodos de visualização de resultados e algumas opções auxiliares.

2 Usando a taxtools

O pacote taxtools.jar pode ser obtido via biblioteca da Embrapa Informática Agropecuária.

Para usar a TaxTools é necessário ter Java (mínimo versão 1.6) e o software estatístico R (mínimo versão 2.11) com a biblioteca “proxy” instalados na máquina.

2.1 Instalando a taxtools

Criar uma nova pasta, por exemplo com o nome “taxtools” e nela colocar o pacote taxtools.jar. Acessar, a partir do terminal de trabalho, esse diretório para poder executar os comandos para as opções da taxtools. Para os exemplos deste tutorial, todos os arquivos de entrada e de saída serão gravados dentro do próprio diretório “taxtools”.

Esse procedimento é o mesmo, tanto para ambientes Linux, quanto para ambientes Windows.

2.2 Instalação do software R com a biblioteca “Proxy”

2.2.1 Em sistemas operacionais Windows

Para fazer o *download* do R, é necessário acessar o site[1] clicar em CRAN e escolher um servidor, de preferência do seu país e mais próximo à sua cidade.

O *download* do R pode ser feito pelas etapas a seguir:

a) clicar em Windows;

b) clicar em “base” para ter acesso à página de onde será feito o *download* do instalador do R;

c) clicar em “Download R X.X for Windows”, onde X.X representa a versão mais recente do R, disponível neste momento.

Por último, aparecerá uma janela, onde se pode executar ou salvar o programa. A forma mais segura de se fazer a instalação é salvar em uma pasta e depois executar o instalador a partir dela, já que o processo de *download* pode ser demorado.

2.2.2 Em sistemas operacionais Linux

Entre com o seguinte comando no terminal:

```
~/ sudo apt-get install r-base
```

[1] *Disponível em: < www.r-project.org>. Acesso em: 16 jul. 2010.*

Após isso, será realizado o *download* automaticamente. Caso não dê certo, no site[2], você encontrará um passo a passo para instalação do R. Para utilizar o R basta digitar “R” em letra maiúscula no terminal e ele será aberto.

2.2.3. Instalação da biblioteca “*proxy*”

Para instalar a biblioteca “*proxy*”, necessária para o cálculo da matriz de dissimilaridade que será utilizada para a construção dos clusters, abra o R, digitando “R” no terminal se o sistema operacional for Linux, ou clicando no atalho se for Windows, e entre com o seguinte comando no prompt do R:

```
> install.packages("proxy")
```

Após isso, será aberta uma janela para escolha de um servidor. Escolha o mais próximo da sua cidade. O R irá fazer *download* da biblioteca que já poderá ser utilizada.

Para chamar a biblioteca no R, utiliza-se o seguinte comando:

```
> library(proxy)
```

3 Parâmetros de entrada

Os parâmetros utilizados pela TaxTools são arquivos no formato Discover (HONORATO; MONARD, 2008), como os gerados pela Pretext2 (SOARES et al., 2008). A Pretext2 é uma ferramenta de pré-processamento de coleções de textos, que permite a análise de coleções em inglês, português e espanhol, e que dá ao usuário liberdade de escolha de várias formas de geração de atributos e filtros. Essa ferramenta também foi desenvolvida no Labic/ ICMC, USP. Para os fins propostos nesse tutorial, utilizaremos como entrada os arquivos “discover.data” e “discover.names”, ambos gerados pela Pretext2.

[2] *Disponível em: <www.r-project.org>. Acesso em: 16 jul. 2010.*

4 Exemplo de arquivos de entrada

Nessa seção, descrevem-se os arquivos de entrada da TaxTools gerados pela Pretext2.

4.1 Discover.data

Nesse arquivo estão os valores dos atributos para todos os textos da coleção. E, ele é equivalente a uma matriz atributoxvalor. Em uma matriz atributoxvalor, cada linha corresponde, nesse a caso, a um documento e cada coluna a um atributo (gerado pela Pretext2). No formato “discover.data”, a primeira coluna contém o caminho e o nome do documento, e cada uma das demais colunas, o valor (tf (frequência do termo), tf-linear (frequência do termo linear), tf-idf (frequência do termo ponderada), booleana) de cada atributo em cada documento; conforme se verifica na Figura 1.

```
"exemplo_Maid/exemplo1.txt",6,0,0,0,0
"exemplo_Maid/exemplo2.txt",1,3,0,0,0
"exemplo_Maid/exemplo3.txt",1,1,1,0,0
"exemplo_Maid/exemplo4.txt",1,1,1,1,0
"exemplo_Maid/exemplo5.txt",5,8,7,5,9
```

Figura 1. Exemplo do arquivo de entrada “discover.data”.

Se houver uma última coluna, com o caminho para um diretório, como no caso da Figura 2, o valor da casela corresponde a um valor de classe (ou rótulo), utilizado por classificadores. A TaxTools, nesse caso, desprezará a última coluna.

```
"exemplo_Maid/exemplo1.txt",6,0,0,0,0,"exemplo_Maid"
"exemplo_Maid/exemplo2.txt",1,3,0,0,0,"exemplo_Maid"
"exemplo_Maid/exemplo3.txt",1,1,1,0,0,"exemplo_Maid"
"exemplo_Maid/exemplo4.txt",1,1,1,1,0,"exemplo_Maid"
"exemplo_Maid/exemplo5.txt",5,8,7,5,9,"exemplo_Maid"
```

Figura 2. Exemplo do arquivo de entrada “discover.data” com rótulo.

4.2 Discover.names

Nesse arquivo está a declaração de todos os atributos da matriz atributo-valor gerada pela Pretext2. A primeira linha do arquivo “discover.names” corresponde ao atributo cujo valor está representado na primeira coluna do arquivo “discover.data”, a segunda linha corresponde ao segundo atributo, e assim por diante. A estrutura do arquivo “discover.names” é exemplificada na Figura 3.

```
filename:string:ignore.
“polit”:integer.
“regul”:integer.
“proced”:integer.
“tranquil”:integer.
“valoriz”:integer.
```

Figura 3. Exemplo do arquivo de entrada “discover.names”.

Se houver uma variável de classe para cada texto, então a primeira linha da “discover.names” será “att_class.”, conforme mostra a Figura 4; os nomes dos atributos de classe aparecem na última linha. A TaxTools, nesse caso, desprezará a primeira e a última linhas desse arquivo.

```
att_class.
filename:string:ignore.
“polit”:integer.
“regul”:integer.
“proced”:integer.
“tranquil”:integer.
“valoriz”:integer.
att_class:nominal(“exemplo_Maid1”,”exemplo_Maid2”).
```

Figura 4. Exemplo do arquivo de entrada “discover.names” com classes.

5 Clusterização

Nesse primeiro passo, a TaxTools recebe os parâmetros de entrada e constrói o *cluster* utilizando a matriz atributo-valor. Para opção de exe-

cução, a TaxTools recebe 4 parâmetros: diretório de trabalho, nome do arquivo “discover.data”, nome do arquivo “discover.names” e tipo de *cluster* (average linkage, single-linkage, complete linkage, centroid linkage, median linkage, Ward) que se deseja fazer. Primeiro a TaxTools invoca o R para realizar o processo de clusterização e, após isso, monta o dendrograma para exibição utilizando programação Java.

A saída dada por essa opção será um dendrograma nomeado como “tipodecluster.hie”, que a TaxTools armazenará no diretório de trabalho.

5.1 Exemplo de uso

Supõe-se, nesse exemplo, que os arquivos de entrada estão na própria pasta da TaxTools e que são nomeados apenas “discover.data” e “discover.names”, geradas a partir da Pretext2, e que deseja-se construir um *cluster* utilizando dissimilaridade de cosseno e a estratégia “*average-linkage*”. Para gerar esses arquivos do tipo Discover foi utilizada uma coleção de 51 textos. Na linha de comando do terminal de trabalho, digita-se:

```
~/taxtools$ java -jar taxtools.jar cluster ./ discover.data discover.
names average
```

Após isso, será exibida uma mensagem de que o R está sendo executado, e quando terminar o processo, outra mensagem avisando que o R foi encerrado e que a função “*dendrogram2hie*” está sendo executada. Ao final do processo, é exibida a mensagem “*Finish*”. No exemplo apresentado, a saída será um dendrograma com nome “*average.hie*”. Caso isso não ocorra, deve-se verificar qual o tipo de erro reportado pelas mensagens.

6 Exibição dos resultados

Para exibir os resultados usa-se a opção *Foldertree*. Ela pode ser utilizada em todas as etapas do processo para que o usuário possa acompanhar o que está sendo feito pelo programa e possa avaliar os resultados passo a passo. Ela tem como entrada 2 parâmetros: o diretório de trabalho e o dendrograma que desejamos visualizar.

Essa opção mostra o dendrograma como uma árvore de arquivos, nela o usuário tem a opção de visualizar quais os atributos em cada ramo da árvore, a frequência com que os atributos aparecem e os documentos contidos no ramo. Além de poder visualizar as medidas de *intracluster*, *intercluster* e de *joinability* para os ramos e para os documentos após calculá-las.

6.1 Exemplo de uso

Visualizando o dendrograma construído a partir do exemplo da opção cluster, portanto, os parâmetros de entrada serão o diretório atual (pasta *dist*) e o arquivo gerado anteriormente "*average.hie*". Na linha de comando do terminal, digita-se:

```
~/taxtools$ java -jar taxtools.jar foldertree ./ ./average.hie
```

Na execução desse comando, será aberta uma janela como a exibida na Figura 5, com a visualização do seu dendrograma.

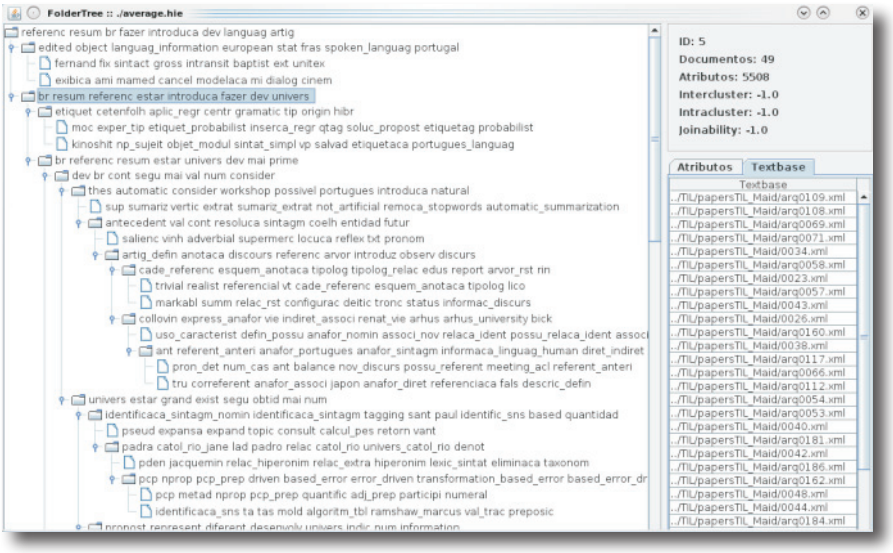

Figura 5. Visualização do dendrograma "*average.hie*".

7 Opção intercluster

Essa opção calcula a medida *intercluster* entre os centroides dos ramos pais e filhos do dendrograma e armazena um novo dendrograma com as medidas já calculadas. Essa medida é usada para avaliar quão significativo é um grupo (*cluster*) dentre os demais grupos do dendrograma; dessa forma, pode-se eliminar grupos não significativos por podas (*pruning*) nas etapas posteriores. Como entrada, a opção recebe: diretório de trabalho, dendrograma de entrada e nome do dendrograma de saída (no qual será gravada a medida calculada).

7.1 Exemplo de uso

Para calcular as medidas *intercluster* no dendrograma "*average.hie*" e depois exibir a *Foldertree* para visualizar-se as medidas calculadas, na linha de comando do terminal, digita-se:

```
~/taxtools$ java -jar taxtools.jar intercluster ./ ./average.hie ./
average-inter.hie
~/taxtools$ java -jar taxtools.jar foldertree ./ ./average-inter.hie
```

A primeira linha de comando usa a opção *intercluster* para calcular as medidas e armazenar no arquivo "*average-inter.hie*". A segunda linha de comando mostra a *foldertree* com as medidas *intercluster* calculadas; essa árvore é mostrada na Figura 6.

8 Opção joinability

Essa opção calcula a medida *joinability* dos *clusters*, que combina os conceitos de *intercluster* e *intracluster*, sendo bastante tolerante à presença de ruídos e à alta dimensionalidade dos dados. Como na opção *intercluster*, essa opção armazena um novo arquivo com o dendrograma com as medidas *joinability* calculadas. Essa medida também pode ser utilizada para podar *clusters* não significativos por métodos de *pruning* em passos seguintes. Como entrada, essa opção de execução recebe: diretório de trabalho, dendrograma de entrada e nome do dendrograma de saída.

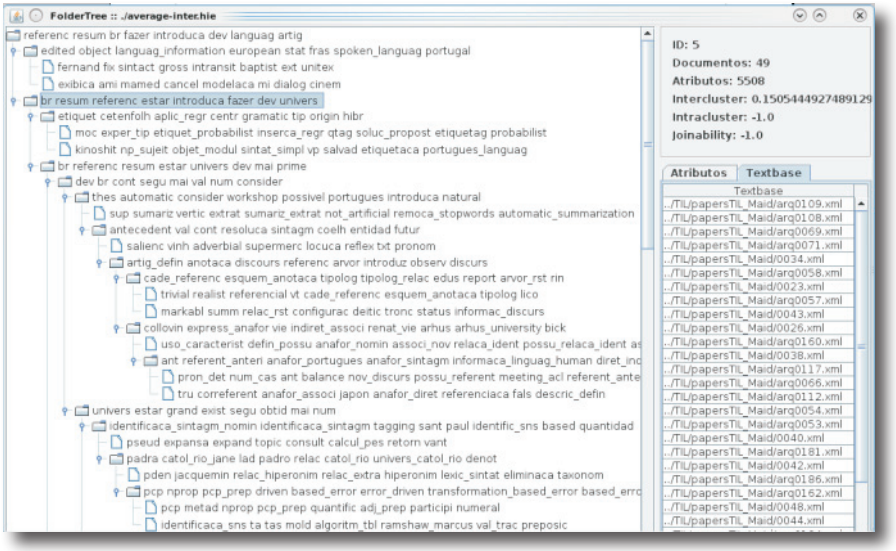

Figura 6. Visualização do dendrograma "*average-inter.hie*" com as medidas inter-cluster calculadas.

8.1 Exemplo de uso

Calcular as medidas *joinability* no dendrograma "average-inter.hie" e depois exibir a *Foldertree* (Figura 7) para visualizar as medidas calculadas. Na linha de comando do terminal, digita-se:

```
~/taxtools$ java -jar taxtools.jar joinability ./ ./average-inter.hie
./average-inter-joinability.hie
~/taxtools$ java -jar taxtools.jar foldertree ./ ./average-inter-
joinability.hie
```

A primeira linha de comando usa a opção *joinability* para calcular as medidas e armazenar no arquivo "average-inter-joinability.hie". E, esse dendrograma armazena as medidas *intercluster* e *joinability* calculadas. A segunda linha de comando mostra a foldertree com as medidas *intercluster* e *joinability* calculadas. E, essa árvore é mostrada na Figura 7.

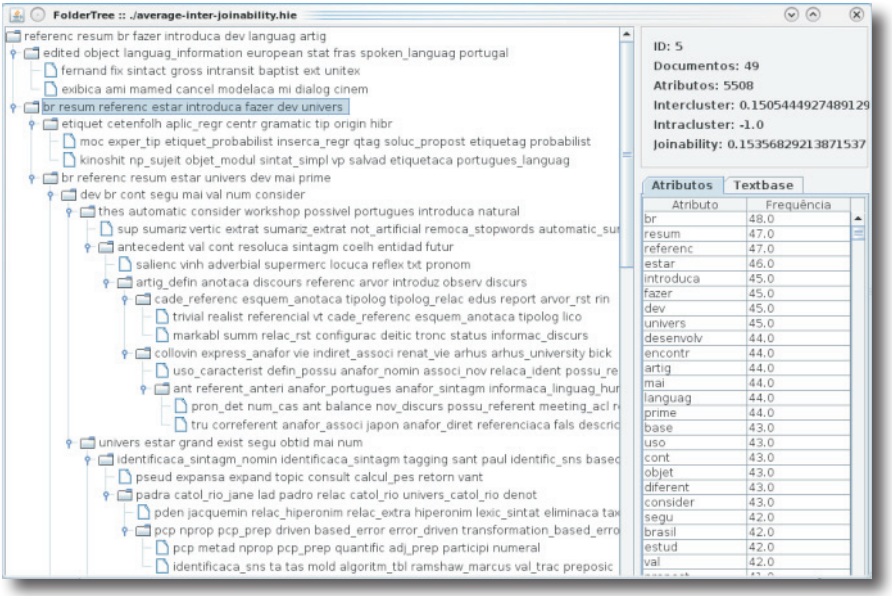

Figura 7. Visualização do dendrograma "*average-inter-joinability.hie*" com as medidas *joinability* calculadas.

9 Opção pruning

Essa opção realiza vários tipos de poda no dendrograma de acordo com os parâmetros passados pelo usuário. Como entrada, essa opção de execução recebe: diretório de trabalho, dendrograma de entrada, nome do dendrograma de saída e tipo de poda que se deseja aplicar ao dendrograma.

A seguir, detalham-se mais claramente alguns tipos de poda, como: 1) mínimo de documentos por *cluster*, 2) poda utilizando a medida *intercluster* e 3) poda utilizando a medida *joinability* (MARCACINI et al., 2009).

9.1 Poda com mínimo de documentos

Esse método de *pruning* elimina os *clusters* que não tenham um número mínimo de documentos estipulado pelo usuário. Como exemplo, realiza-se essa poda no dendrograma "*average-inter-joinability.hie*" com um mínimo

de 5 documentos por *cluster* e depois exibi-se sua *foldertree* para visualizar o resultado. Na linha de comando do terminal, digita-se:

```
~/taxtools$ java -jar taxtools.jar pruning ./ ./average-inter-joinability.hie ./average-inter-joinability-min5.hie mincut 5
~/taxtools$ java -jar taxtools.jar foldertree ./ ./average-inter-joinability-min5.hie
```

A primeira linha de comando usa a opção *pruning* para podar os *clusters* com menos de 5 documentos e armazenar no arquivo "*average-inter-joinability-min5.hie*",. A segunda linha de comando mostra a *foldertree* com as medidas *intercluster* e *joinability* calculadas. E, essa árvore é mostrada na Figura 8.

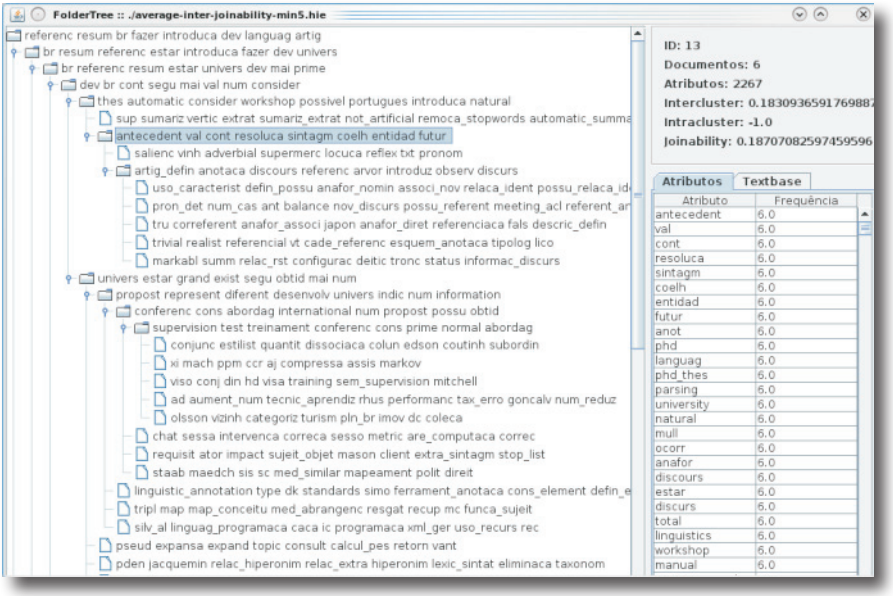

Figura 8. Visualização do dendrograma "*average-inter-joinability-min5.hie*" após realização da poda com mínimo de documentos.

9.2 Poda com intercluster

Esse método de *pruning* poda os *clusters* de acordo com uma medida mínima de *intercluster* estipulada pelo usuário. Como exemplo, realiza-se

essa poda no dendrograma "*average-inter-joinability-min5.hie*" com uma medida mínima de *intercluster* de 0,2 e depois exibi-se sua *foldertree* para visualizar o resultado. Na linha de comando do terminal, digita-se:

```
~/taxtools$ java -jar taxtools.jar pruning ./ ./average-inter-joina-
bility-min5.hie ./average-inter0.2-joinability-min5.hie intercluster
0.2
~/taxtools$ java -jar taxtools.jar foldertree ./ ./average-inter0.2-
joinability-min5.hie
```

A primeira linha de comando usa a opção *pruning* para podar os *clusters* com medidas de *intercluster* menores que 0.2 e armazenar no arquivo "*average-inter0.2-joinability-min5.hie*",. A segunda linha de comando mostra a *foldertree* com as medidas *intercluster* e *joinability* calculadas. E, essa árvore é mostrada na Figura 9.

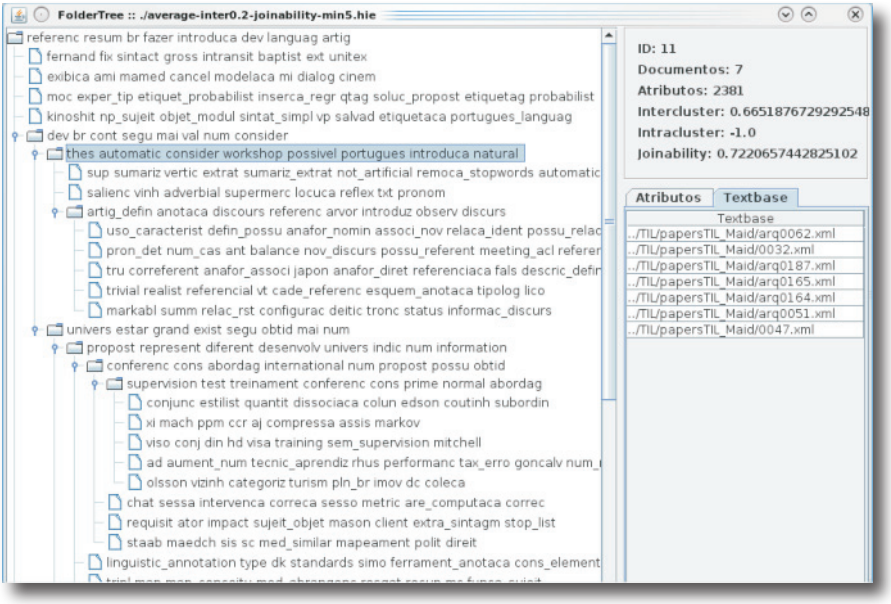

Figura 9. Visualização do dendrograma "*average-inter0.2-joinability-min5.hie*" após realização da poda com *intercluster*.

9.3 Poda com joinabilitY

Esse método de *pruning* poda os *clusters* de acordo com uma medida mínima de *joinability* estipulada pelo usuário. Por exemplo, para realizar essa poda no dendrograma "*average-inter0.2-joinability-min5.hie*" com uma medida mínima de *joinability* de 0,3 e depois exibir sua *foldertree* para visualizar o resultado. Na linha de comando do terminal, digita-se:

```
~/taxtools$ java -jar taxtools.jar pruning ./ ./average-inter0.2-joinability-min5.hie ./average-inter0.2-joinability0.3-min5.hie joinability 0.3
~/taxtools$ java -jar taxtools.jar foldertree ./ ./average-inter0.2-joinability0.3-min5.hie
```

A primeira linha de comando usa a opção *pruning* para cortar os *clusters* com medidas de joinability menores que 0.3 e armazenar no arquivo "*average-inter0.2-joinability0.3-min5.hie*". A segunda linha de comando mostra a *foldertree* com as medidas *intercluster* e *joinability* calculadas. E, essa árvore é mostrada na Figura 10.

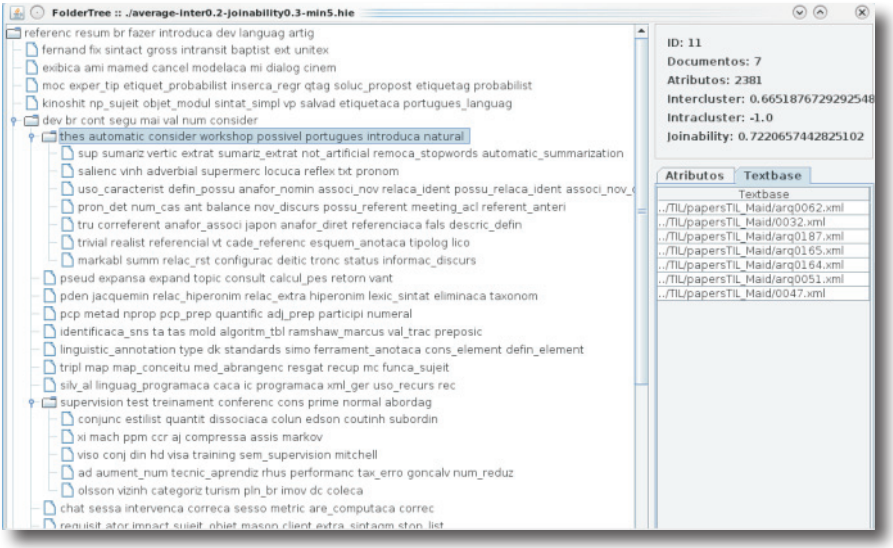

Figura 10. Visualização do dendrograma "*average-inter0.2-joinability0.3-min5.hie*" após realização da poda com *joinability*.

10 Opção explorer

A opção *explorer* é uma outra forma de visualização, que mostra a árvore em formato de pastas e sub-pastas no *frame* esquerdo. No *frame* superior direito são exibidos os documentos contidos no ramo selecionado e no *frame* inferior direito é exibido o texto do documento selecionado. Essa opção recebe como entrada três parâmetros: diretório de trabalho, árvore a ser exibida e diretório onde estão contidos os textos da coleção de documentos.

10.1 Exemplo de uso

Supõe-se, para esse exemplo, que a pasta contendo os textos é uma subpasta da própria pasta *dist*, e que ela é a pasta *Maid* gerada pela Pretext2. Quando se usa a Pretext2 para o pré-processamento da coleção de textos, ela cria uma pasta *Maid*, na qual são guardados os textos após o processamento. Suponha que se queira visualizar o arquivo "*average-inter0.2-joinability0.3-min5.hie*", para isso, digita-se na linha de comando do terminal:

```
~/taxtools$ java -jar taxtools.jar explorer ./ ./average-inter0.2-joinability0.3-min5.hie ./
```

Após entrar com o comando, será aberta uma janela como a exibida na Figura 11, com a visualização do seu dendrograma.

11 Opção GRAPH

A opção *graph* também corresponde a uma forma de visualização, onde o dendrograma é exibido em forma de grafo. Ela mostra os ramos com os atributos e é interativa: quando o usuário passa o *mouse* sobre um dos ramos, este fica vermelho e os ramos ligados a ele ficam laranjas. O usuário pode também controlar uma série de opções que mudam o formato do grafo e o modo como ele se comporta. Como entrada, essa opção recebe dois atributos: diretório de trabalho e dendrograma a ser exibido.

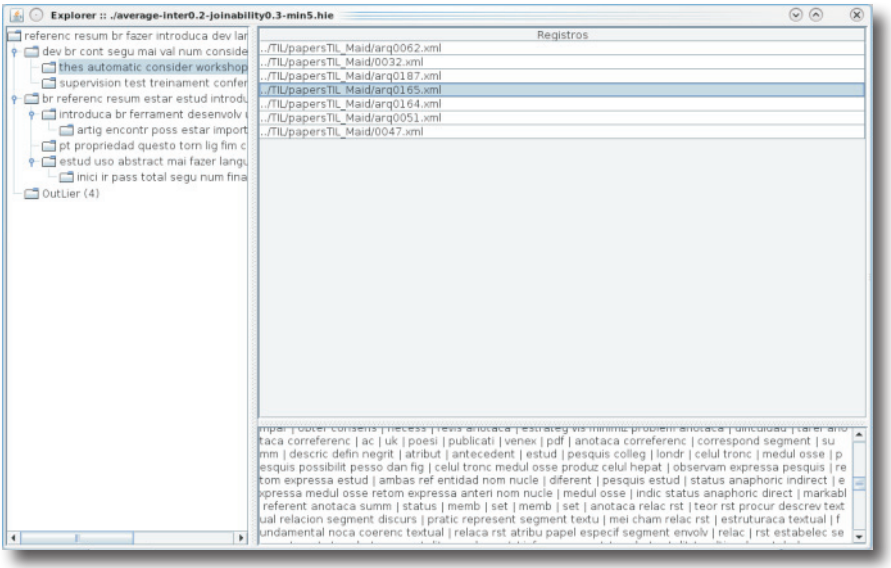

Figura 11. Visualização do dendrograma "*average-inter0.2-joinability0.3-min5.hie*" pela opção explorer.

11.1 Exemplo de uso

Para ilustrar o uso da opção *graph*, exibi-se o dendrograma "*average-inter0.2-joinability0.3-min5.hie*". Para isso, digita-se na linha de comando do terminal:

```
~/taxtools$ java -jar taxtools.jar graph ./ ./average-inter0.2-
joinability0.3-min5.hie
```

Após isso, a opção abrirá uma janela como a exibida na Figura 12, com a visualização do seu dendrograma em forma de grafo.

12 Opção RLUM

O Robust Labelling Up Method (RLUM) é um método de geração de rótulos para o agrupamento de documentos textuais (MOURA; REZENDE, 2010), que visam à identificação de palavras-chaves para indicar os possíveis

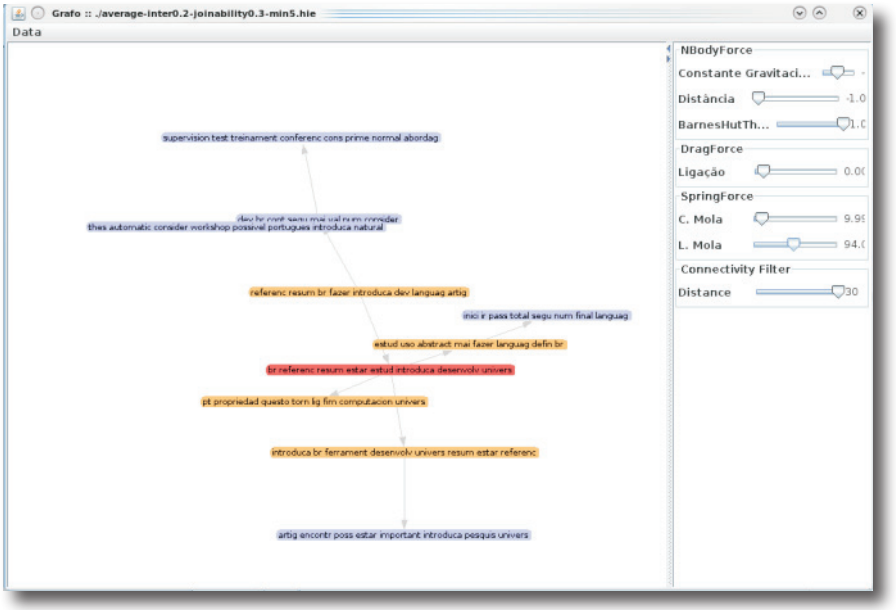

Figura 12. Visualização do dendrograma "*average-inter0.2-joinability0.3-min5.hie*" em forma de grafo.

tópicos aos quais os documentos agrupados se refiram. A opção RLUM aplica esse método ao dendrograma após os cortes terem sido realizados. Após a aplicação do método não será possível realizar novos cortes.

12.1 Exemplo de uso

Para ilustrar o uso da opção RLUM, aplica-se o método no dendrograma "*average-inter0.2-joinability0.3-min5.hie*". Para isso, digita-se na linha de comando do terminal:

```
~/taxtools$ java -jar taxtools.jar rlum ./ ./average-inter0.2-
joinability0.3-min5.hie ./average-inter0.2-joinability0.3-min5-rlum.
hie
~/taxtools$ java -jar taxtools.jar foldertree ./ ./average-inter0.2-
joinability0.3-min5-rlum.hie
```

A primeira linha de comando usa a opção rlum para gerar a rotulação da coleção e armazenar no arquivo "*average-inter0.2-joinability0.3-min5-rlum.*

hie". A segunda linha de comando mostra a *foldertree* com as medidas *intercluster* e *joinability* calculadas. E, essa árvore é mostrada na Figura 13.

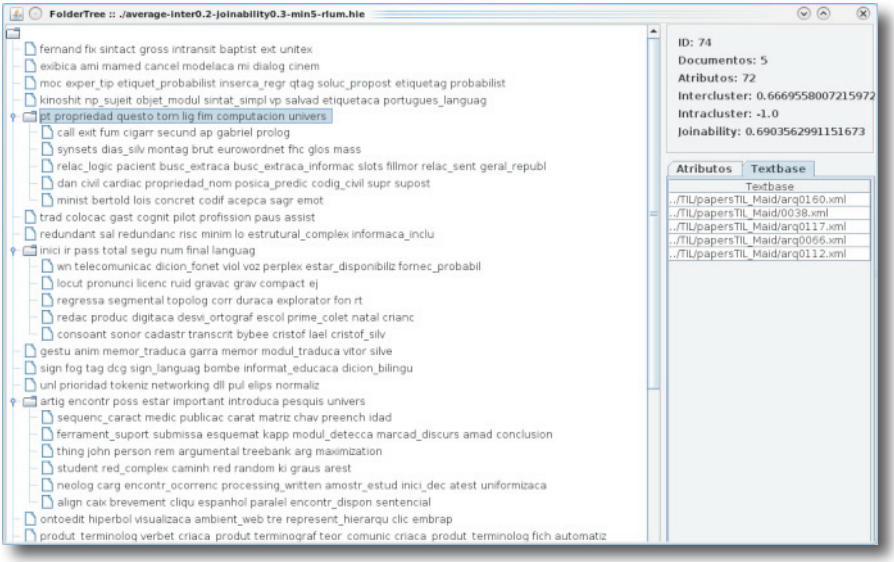

Figura 13. Visualização do dendrograma "*average-inter0.2-joinability0.3-min5-rlum.hie*" após realização da rotulação.

13 Opção RLDM

O Robust Labelling Down Method (RLDM) é um método de geração de rótulos para o agrupamento de documentos textuais (MOURA; REZENDE, 2010), que visam à identificação de palavras-chaves para indicar os possíveis tópicos aos quais os documentos agrupados se referiram. Esse método permite que seu algoritmo sempre tome uma decisão em relação a algum a_k (atributo) e a em algum n_i (nó da árvore) e, consequentemente, os algoritmos sempre produzem um conjunto pequeno de rótulos em cada grupo e não replicam os termos ao longo da hierarquia. A opção RLDM aplica esse método ao dendrograma após os cortes terem sido realizados. Após a aplicação do método não será possível realizar novos cortes.

13.1 Exemplo de uso

Para ilustrar o uso da opção RLDM, aplica-se o método no dendrograma "*average-inter0.2-joinability0.3-min5.hie*". Para isso, digita-se na linha de comando do terminal:

```
~/taxtools$ java -jar taxtools.jar rldm ./ ./average-inter0.2-
joinability0.3-min5.hie ./average-inter0.2-joinability0.3-min5-rldm.
hie
~/taxtools$ java -jar taxtools.jar foldertree ./ ./average-inter0.2-
joinability0.3-min5-rldm.hie
```

A primeira linha de comando usa a opção *rldm* para gerar a rotulação da coleção e armazenar no arquivo "*average-inter0.2-joinability0.3-min5-rldm.hie*". A segunda linha de comando mostra a *foldertree* com as medidas *intercluster* e *joinability* calculadas. E, essa árvore é mostrada na Figura 14.

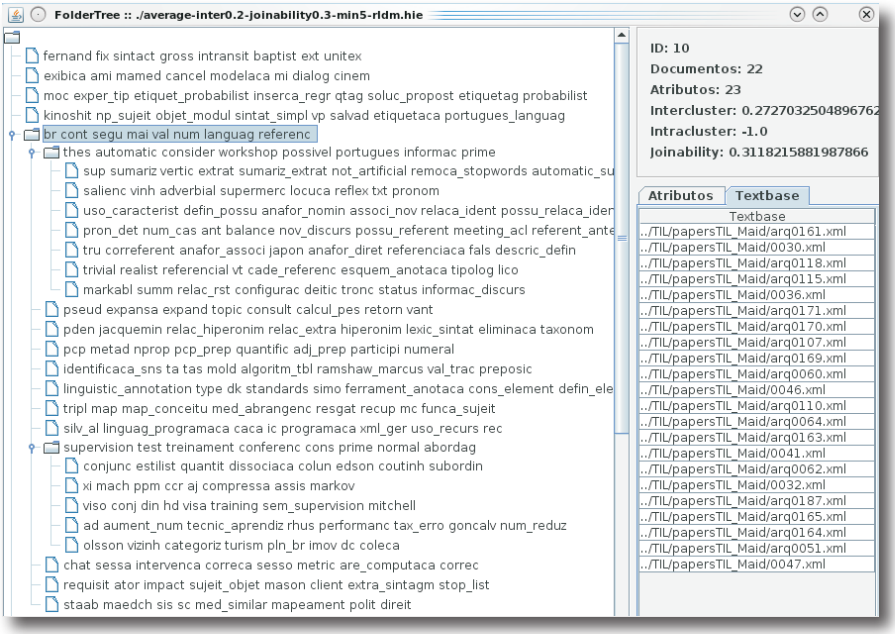

Figura 14. Visualização do dendrograma "*average-inter0.2-joinability0.3-min5-rldm.hie*" após realização da rotulação.

Referências

GANTZ, J.; REINSEL, D. “**The digital universe decade**: are you ready?” IDC white paper, May 2010. Disponível em: < http://idcdocserv.com/925>. Acesso em: 20 dez. 2010.

HONORATO, D. F., MONARD, M. C. **Descrição do ambiente computacional tp-Discover para mapear informação não estruturada em uma Tabela Atributo-Valor**. USP, ICMC, São Carlos, SP, 2008. (Relatório técnico, 318).

MARCACINI, R. M.; MOURA, M. F.; REZENDE, S. O. Uma abordagem para seleção de grupos significativos em agrupamento hierárquico de documentos. In: ENCONTRO NACIONAL DE INTELIGÊNCIA ARTIFICIAL, 2009, Bento Gonçalves. **Anais**... Porto Alegre: UFRGS, 2009. p. 1-16. 1 CD-ROM.

MOURA, M. F.; REZENDE, S. O. A simple method for labeling hierarchical document clusters. In: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE AND APPLICATIONS, 10., 2010, Innsbruk – Austria. **Proceedings**... Acta Press, 2010. v. 1, p. 336-371.

MOURA, M. F.; MARCACINI, R. M.; NOGUEIRA, B. M.; CONRADO, M. S.; REZENDE, S. O. A proposal for building domain topic taxonomies. In: WORKSHOP ON WEB AND TEXT INTELLIGENCE, 1.; SIMPÓSIO BRASILEIRO DE INTELIGÊNCIA ARTIFICIAL, 19., 2008, Salvador. **Proceedings**... São Carlos, SP : USP, ICMC, 2008. v. 1, p. 83-84.

SOARES, M. V. B., PRATI, R. C.; MONARD, M. C. **PreTexT II**: descrição da reestruturação da ferramenta de pré-processamento de textos. São Carlos, SP: USP, CMC 2008. 45 p. (Relatório técnico, 333). Disponível em: <http://www.icmc.usp.br/~biblio/BIBLIOTECA/rel_tec/RT_333.pdf>. Acesso em: 16 jul. 2010.

Embrapa

Informática Agropecuária

Ministério da
Agricultura, Pecuária
e Abastecimento

BRASIL
UM PAÍS DE TODOS
GOVERNO FEDERAL

CGPE 9066