

**UNIVERSIDADE DE SÃO PAULO**  
**Instituto de Ciências Matemáticas e de Computação**  
ISSN 0103-2569

---

**Avaliação de Ambientes de Suporte à Montagem  
Automática de Córpus a Partir de Textos da Web e  
Extração Automática de Termos**

**Luiz Carlos Genovês Jr.  
Sandra Maria Aluisio**

**Nº 266**

---

**RELATÓRIOS TÉCNICOS**



**São Carlos – SP  
Out./2005**

|             |         |
|-------------|---------|
| SYSNO       | 1474327 |
| DATA        | / /     |
| ICMC - SBAB |         |

Universidade de São Paulo - USP  
Universidade Federal de São Carlos - UFSCar  
Universidade Estadual Paulista - UNESP

**Avaliação de ambientes de suporte à montagem  
automática de corpus a partir de textos da Web e extração  
automática de termos**

Luiz Carlos Genovês Jr.  
Sandra Maria Aluisio

**NILC-TR-05-15**

Outubro 2005

Série de Relatórios do Núcleo Interinstitucional de Linguística Computacional  
NILC - ICMC-USP, Caixa Postal 668, 13560-970 São Carlos, SP, Brasil

## Conteúdo

|                                                                                                                                                                                      |    |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Índice de Figuras.....                                                                                                                                                               | 3  |
| Índice de Tabelas .....                                                                                                                                                              | 4  |
| Resumo.....                                                                                                                                                                          | 5  |
| 1. Introdução .....                                                                                                                                                                  | 6  |
| 1.1. Contextualização, Motivação e Domínio de Aplicação.....                                                                                                                         | 6  |
| 1.2. Objetivos do Trabalho.....                                                                                                                                                      | 9  |
| 1.3. Organização da Monografia .....                                                                                                                                                 | 9  |
| 2. Revisão Bibliográfica .....                                                                                                                                                       | 10 |
| 2.1. Síntese das Tecnologias Envolvidas.....                                                                                                                                         | 10 |
| 2.1.1. O Ambiente Corpógrafo.....                                                                                                                                                    | 10 |
| 2.1.2. O ToolKit BootCaT .....                                                                                                                                                       | 14 |
| 2.1.3. A API do Google .....                                                                                                                                                         | 16 |
| 2.1.4. Extração automática de termos .....                                                                                                                                           | 17 |
| 2.2. Trabalhos Relacionados .....                                                                                                                                                    | 18 |
| 2.3. Análise Crítica.....                                                                                                                                                            | 19 |
| 3. Estado atual do trabalho .....                                                                                                                                                    | 21 |
| 3.1. Projeto.....                                                                                                                                                                    | 21 |
| 3.1.1. Extração de unigramas e termos multi-palavras e comparação entre métodos.....                                                                                                 | 21 |
| 3.1.2. Suporte à extração automática de córpus.....                                                                                                                                  | 22 |
| 3.2. Equipe .....                                                                                                                                                                    | 22 |
| 3.3. Descrição das Atividades Realizadas .....                                                                                                                                       | 23 |
| 3.3.1. Avaliação dos scripts do BootCaT: levantando suas limitações .....                                                                                                            | 23 |
| 3.3.2. Inclusão de novas ferramentas para conversão de tipos de arquivo.....                                                                                                         | 23 |
| 3.3.3. Extração de unigramas usando o BootCaT .....                                                                                                                                  | 24 |
| 3.3.4. Extração de bigramas e trigramas usando o BootCaT .....                                                                                                                       | 24 |
| 3.4. Resultados Obtidos .....                                                                                                                                                        | 26 |
| 3.4.1. Extração de unigramas e termos multi-palavras .....                                                                                                                           | 26 |
| 3.4.2. Suporte à montagem de córpus .....                                                                                                                                            | 28 |
| 3.5. Dificuldades e Limitações.....                                                                                                                                                  | 28 |
| 4. Conclusões e Trabalhos Futuros .....                                                                                                                                              | 29 |
| Referências Bibliográficas.....                                                                                                                                                      | 31 |
| Apêndice 1: Resumo das ferramentas de obtenção de córpus do BootCaT .....                                                                                                            | 32 |
| Apêndice 2: Recorte da Ontologia sobre Nanociência e Nanotecnologia .....                                                                                                            | 34 |
| Apêndice 3: Geração das listas de unigramas, bigramas e trigramas .....                                                                                                              | 38 |
| Apêndice 4: Lista de Stopwords .....                                                                                                                                                 | 43 |
| Anexo 1: Ambiente Web colaborativo do Projeto <i>Extração automática de termos e elaboração colaborativa de terminologias para o intercâmbio de conhecimento especializado</i> ..... | 47 |
| Anexo 2: Interface Web do Ambiente e-Termos apresentado o Módulo de criação de córpus decartáveis implementado com o BootCat .....                                                   | 48 |
| Glossário .....                                                                                                                                                                      | 51 |

## Índice de Figuras

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |    |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figura 1: Roteiro completo do Corpógrafo.....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | 12 |
| Figura 2: Funcionalidades do Gestor de Ficheiros do Corpógrafo .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 13 |
| Figura 3: Fluxo de montagem de um córpus no BootCaT [4] .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | 14 |
| Figura 4: Estrutura de um elemento retornado por uma busca feita pela API do Google .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 17 |
| Figura 5: Estrutura do Ambiente Web Colaborativo .....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 47 |
| Figura 6: Tela inicial do Processo de Geração de um Córpus Descartável para Pesquisas Terminológicas. Essa tela permite fazer o upload de um arquivo com semestres (os autores do BootCat recomendam de 5 a 15 sementes que são termos discriminantes do domínio de pesquisa); no exemplo da figura foram escolhidas 15 sementes e pedida a combinação 2 a 2 com elas -- o que daria 105 combinações no máximo -- e para retornar 15 combinações desse conjunto de 105. O domínio escolhido foi o de Revestimentos Cerâmicos. Na segunda janela (“Combinações”) são apresentadas as 15 combinações com 2 termos escolhidas e na terceira janela (“Lista de Consultas”) as combinações que serão passadas como buscas ao Google. Escolhe-se também a quantidade de links a serem gerados para cada consulta. No caso da figura foram escolhidos 2 links..... | 48 |
| Figura 7. Segunda tela do Processo de Geração de um Córpus Descartável para Pesquisas Terminológicas. Essa tela mostra o total de links retornados e o número de duplicados. Pede o nome do córpus a ser criado e a língua. Ao clicar no botão “Obter Textos” o processo de obtenção de textos da Web com o Google se inicia.....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | 49 |
| Figura 8: Córpus compilado com sucesso. O córpus Rev_ceramicos fica então disponível para os próximos módulos do ambiente e-Termos mostrado esquematicamente no Anexo 1. ....                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | 50 |

## Índice de Tabelas

|                                                                                                                                             |    |
|---------------------------------------------------------------------------------------------------------------------------------------------|----|
| Tabela 1: Candidatos a bigramas .....                                                                                                       | 25 |
| Tabela 2: Candidatos a trigramas.....                                                                                                       | 25 |
| Tabela 3: Número dos candidatos a termos obtidos pelos métodos do BootCaT, Frequência e da ontologia .....                                  | 26 |
| Tabela 4: Número de candidatos a termos obtido das redes complexas.....                                                                     | 26 |
| Tabela 5: Comparação entre os termos obtidos utilizando o BootCaT e a ontologia.....                                                        | 27 |
| Tabela 6: Comparação entre os termos obtidos utilizando a frequência e a ontologia.....                                                     | 27 |
| Tabela 7: Comparação entre os unigramas obtidos utilizando diferentes métricas da redes complexas e a ontologia.....                        | 27 |
| Tabela 8: Comparação entre os termos obtidos utilizando o BootCaT e os obtidos pelo corte na frequência.....                                | 28 |
| Tabela 9: Comparação entre os unigramas obtidos utilizando diferentes métricas da redes complexas e os unigramas obtidos pelo BootCaT ..... | 28 |
| Tabela 10: Conjunto de ferramentas de PLN.....                                                                                              | 47 |

## Resumo

A utilização de *corpus* está se generalizando progressivamente nas áreas da Lingüística, do Processamento de Língua Natural (PLN), Tradução e Terminologia, para citar algumas. Embora já exista um número razoável de *corpus* genéricos (ou de referência, como são chamados) para várias línguas, o número de *corpus* específicos disponíveis para suporte à pesquisa terminológica, atividades de tradução e avaliação de recursos de PLN ainda é deficiente. Enquanto é possível construir tais *corpus* por meio de busca manual na Web esse processo consome muito tempo se levarmos em conta os benefícios para pesquisas únicas. Para atender essa necessidade específica de criação e de pesquisa nesses tipos de *corpus* existem vários projetos que se propõem a disponibilizar ambientes para criação rápida de *corpus* e posterior pesquisa com eles. Um desses projetos é o Corpógrafo e outro o BootCaT. Os objetivos desta pesquisa foram o estudo e a avaliação dos dois ambientes acima mencionados para gerenciamento de *corpus* especializados, para serem utilizados, sejam suas próprias ferramentas ou a funcionalidade delas, no projeto *Portal da Rede de Nanotecnologia da USP* e no projeto *Extração automática de termos e elaboração colaborativa de terminologias para o intercâmbio de conhecimento especializado*, ambos sendo desenvolvidos no Núcleo Interinstitucional de Lingüística Computacional.

## 1. Introdução

### 1.1. Contextualização, Motivação e Domínio de Aplicação

A utilização de *corpus*<sup>1</sup> está se generalizando progressivamente nas áreas da Linguística, do Processamento de Língua Natural (PLN), Tradução e Terminologia, para citar algumas. Segundo Garside, Leech & MacEnery [1], o termo *corpus* designa um conjunto de dados linguísticos autênticos que pode consistir de textos escritos, fala transcrita ou ambos. Nos últimos 35 anos, o conceito de *corpus* tem sido usado como sinônimo de “*corpus* computadorizado”, isto é, os dados estão em formato eletrônico, podendo ser processados por computador e servir a vários propósitos tais como a pesquisa linguística ou o processamento de língua natural. É possível enriquecer um *corpus* com o uso de anotação linguística em vários níveis: morfossintático, sintático, semântico e retórico, por exemplo. Essas anotações agregam valor ao *corpus* e permitem que várias outras pesquisas sejam feitas nesses dados. Por exemplo, em um *corpus* com anotação morfossintática pode-se procurar por padrões de termos, como “substantivo–adjetivo” e “substantivo–preposição–substantivo”.

Nos últimos anos, tem havido um grande esforço na criação de grandes *corpus*, constituídos por vários milhões de palavras, para fins genéricos de pesquisa, e na sua posterior disponibilização, tanto em formato CD-ROM como por meio de ambientes de acesso via Web. Por exemplo, temos o BNC<sup>2</sup> e o ANC<sup>3</sup> para a língua inglesa nas variantes britânica e americana, respectivamente, e o CETEMPúblico<sup>4</sup> e o Lácio-Web<sup>5</sup>, para a língua portuguesa nas variantes europeia e brasileira, respectivamente. Esse último foi desenvolvido num consórcio entre o NILC<sup>6</sup>/ICMC-USP, FFLCH-USP, IME-USP e disponibilizado em julho de 2004.

---

<sup>1</sup> Neste trabalho, utilizamos o aportuguesamento da palavra *corpus* (plural *corpora*) tendo a mesma ortografia para o plural e singular.

<sup>2</sup> <http://www.natcorp.ox.ac.uk/>

<sup>3</sup> <http://americannationalcorpus.org/>

<sup>4</sup> <http://www.linguatca.pt/>

<sup>5</sup> <http://www.nilc.icmc.usp.br/lacioweb/>

<sup>6</sup> <http://www.nilc.icmc.usp.br/nilc/index.html>

Embora já exista um número razoável de *córpus* genéricos (ou de referência, como são chamados) para várias línguas, o número de *córpus* específicos disponíveis para suporte à pesquisa terminológica, atividades de tradução e avaliação de recursos de PLN ainda é deficiente. Essa deficiência dá-se pela própria especificidade de tais *córpus* que são muitas vezes construídos para serem utilizados por um período curto de tempo e por isso se questiona o investimento de grandes esforços na sua compilação e anotação, seja essa de cabeçalho (informações bibliográficas e outras sobre os textos) e lingüística. Além disso, pode acontecer de tais *córpus* específicos serem utilizados somente em um projeto o que representa uma perda dos recursos dispensados em sua compilação.

Enquanto é possível construir tais *córpus* pela busca manual na Web, esse processo consome muito tempo se levarmos em conta os benefícios para pesquisas únicas. Para atender essa necessidade específica de criação e de pesquisa nesses tipos de *córpus*, existem vários projetos que se propõem a disponibilizar ambientes para criação rápida de *córpus* e posterior pesquisa, tratamento e disponibilização dos resultados do trabalho, por exemplo uma lista de candidatos a termos em pesquisas terminológicas. Um desses projetos é o Corpógrafo [2,3] e outro o BootCaT [4].

O Corpógrafo é um ambiente Web integrado que dá suporte a várias tarefas: montagem da coleção de textos da Web (que podem estar em vários formatos: PDF, HTML, Word, PS), limpeza dos textos, busca utilizando expressão regular, concordância, extração de colocações e contagem de *n*-gramas<sup>7</sup>, extração de candidatos a termos e relações semânticas, compilação de produtos terminológicos, como glossários e tesouros, e exportação desses para outros formatos e aplicações. Já o BootCaT é um conjunto de ferramentas perl implementando um procedimento iterativo para a construção de *córpus* especializados e candidatos a termos da Web e requer somente uma lista pequena de candidatos (termos típicos do domínio) como entrada. Atualmente, permite somente a inclusão de textos no formato em HTML, mas essa restrição pode ser contornada com a inclusão de rotinas para tratamento dos formatos mais comuns de textos da Web.

Além desses dois ambientes, existe um projeto aprovado pela FAPESP em novembro de 2003 para a construção de um outro ambiente. O projeto é intitulado *Extração automática de termos e elaboração colaborativa de terminologias para o intercâmbio de conhecimento*

---

<sup>7</sup> termos com *n* de gramas. Exemplo: para *n* = 2, bigramas; para *n* = 1, unigramas.

*especializado* (processo nº. 2003/06569-3), e está sendo desenvolvido com a coordenação da Profa. Dra. Gladis Maria de Barcellos Almeida do Departamento de Letras (DL) da Universidade Federal de São Carlos (UFSCar) e Profa Sandra Maria Aluísio do ICMC-USP (ambas são coordenadoras do NILC - Núcleo Interinstitucional de Linguística Computacional). Ele visa propor soluções para a pesquisa terminológica por meio de ferramentas computacionais baseadas na Web e compartilha da mesma motivação dos ambientes acima, mas possui particularidades quanto aos algoritmos de extração, visualização e edição de produtos terminológicos e formato de exportação dos resultados. Para a criação desse ambiente de apoio à criação de produtos terminológicos a partir de *cópus*, existe um projeto de doutorado no ICMC orientado pela mesma orientadora deste projeto de graduação. Entretanto, o projeto original não previa a criação do *cópus* de pesquisa a partir de textos da Web suportada por ferramentas nos mesmos moldes das do BootCaT. No projeto Fapesp original e também no de doutorado que o encampou partia-se de um *cópus* já construído.

Existe também outro projeto sendo desenvolvido no ICMC, especificamente no NILC, relacionado com o *Portal da Rede de Nanotecnologia da USP*<sup>8</sup>. Esse projeto prevê um estudo sobre terminologia para estabelecer uma estrutura conceitual (ontologia<sup>9</sup>) para a nanotecnologia, que possa não apenas fornecer subsídios para produzir um Portal abrangente e de alta qualidade, mas também guiar a busca de oportunidades de mercado e oferta de tecnologias. Para a construção da ontologia, o projeto previa o uso de métodos automáticos de extração de termos a partir de *cópus* específicos de uma área do conhecimento. Esse cenário permitiu avaliação de vários métodos de extração a partir de um grande *cópus* criado com textos da área de nanociência e nanotecnologia.

O domínio de aplicação deste projeto de graduação se insere nas áreas de pesquisa em Linguística de *Cópus* e Terminologia e prevê a utilização de ferramentas para construção de *cópus* a partir de textos da Web e de métodos automáticos de extração de termos (tanto os termos aqueles formados por palavras únicas como por multi-palavras). Ele colabora com os dois projetos citados acima da seguinte forma: para o projeto FAPESP, ele prevê a criação de um módulo extra que auxilia na criação de textos a partir da Web, e para o Portal da Rede de

---

<sup>8</sup> <http://www.usp.br/prp/nanotecnologia/>

<sup>9</sup> Neste trabalho, ontologia é entendida como uma estrutura hierárquica com as relações de inclusão de classes (*é um tipo de*) e instanciação que permite a inclusão dos termos do domínio.

Nanotecnologia, prevê a avaliação de um método de extração automática de termos que é disponibilizado no toolkit BootCaT com outros dois métodos sendo utilizados no projeto.

## **1.2. Objetivos do Trabalho**

Os objetivos desta pesquisa foram: o estudo e avaliação dos dois ambientes acima mencionados para gerenciamento de corpúscos especializados (BootCaT e Corpógrafo) para serem utilizados, sejam suas próprias ferramentas ou a funcionalidade delas, no projeto *Portal da Rede de Nanotecnologia da USP* e no projeto *Extração automática de termos e elaboração colaborativa de terminologias para o intercâmbio de conhecimento especializado*. Especificamente, o projeto de graduação avaliou o método de extração automática de termos do BootCaT com um corpúscos criado no âmbito do projeto e o comparou com outros dois métodos utilizados por outros membros, usando as medidas de precisão e revocação contra uma ontologia (parcial) composta de termos advindos do corpúscos. Também, projetou o módulo de criação de textos a partir da Web, com ferramentas do BootCaT, o qual foi implementado por outro membro do projeto.

## **1.3. Organização da Monografia**

O restante da monografia está organizado como segue. No Capítulo 2 são apresentadas as funcionalidades do Corpógrafo e do toolkit BootCat, além de uma descrição breve da API (Interface para Programação de Aplicativos) do Google, pois embasaram o trabalho no projeto *Extração automática de termos e elaboração colaborativa de terminologias para o intercâmbio de conhecimento especializado*. Também apresentamos a metodologia utilizada para construção da ontologia para a área de nanotecnologia e a metodologia utilizada na comparação entre os métodos de extração de termos, juntamente com uma breve descrição destes métodos. Finalmente, explicamos porque este cenário motivou o presente projeto. No Capítulo 3, o projeto é descrito mais detalhadamente, bem como a equipe dos projetos maiores que colaboraram e são apoiadas por este trabalho. Todas as atividades feitas estão reportadas neste capítulo, assim como os resultados aferidos e as dificuldades encontradas no

trajeto. No quarto e último capítulo, descrevemos as conclusões deste trabalho, explicitando suas limitações e os trabalhos futuros.

## **2. Revisão Bibliográfica**

### **2.1. Síntese das Tecnologias Envolvidas**

#### **2.1.1. O Ambiente Corpógrafo**

Desenvolvido pela Faculdade de Letras da Universidade do Porto (FLUP), o Corpógrafo<sup>10</sup> é um gestor de córpus que se encontra, atualmente, direcionado para pesquisas terminológicas, isto é, a extração de termos e organização deles em bases de dados. Fornece um ambiente Web integrado para o manejo de córpus, disponibilizando ferramentas para processamento de córpus. Dentre as ferramentas que possui, estão concordanceadores, contadores de frequência, e também ferramentas de pré-processamento de córpus, como as de limpeza de córpus e sentenciadores. Toda funcionalidade do Corpógrafo está associada a um dos quatro ambientes de trabalho ou módulos: gestor de ficheiros, pesquisa de corpora, centro de conhecimento, centro de documentação, diminuindo a sobrecarga ao trabalhar no ambiente (Figura 1).

Dos quatro módulos contidos no Corpógrafo, o que mais interessou no estudo feito neste projeto foi o “Gestor de ficheiros” (Figura 2), que trata especificamente da montagem de córpus. Para construir um córpus no Corpógrafo, primeiramente é necessário selecionar os textos que comporão o córpus, que podem ser fornecidos de duas maneiras: ou enviando o próprio arquivo (*upload*) ou informando a URL onde o arquivo pode ser encontrado. O Corpógrafo aceita textos do tipo PDF, HTML, DOC, PS e RTF, além do TXT, formato para o qual todos os outros tipos de texto são transformados. O Corpógrafo oferece ferramentas para o pré-processamento desses textos, tais como sentenciadores (denominados “fraseadores” em português de Portugal) e um ambiente de edição que permite fazer a “limpeza” de textos (retirar lixo provindo da conversão de

---

<sup>10</sup> <http://www.linguateca.pt/Corpografo/>

tipos de texto, remoção de cabeçalhos, tabelas, referências ou agradecimentos). Após pré-processar os textos, pode-se selecionar aqueles que farão parte do corpus.

Tendo um corpus montado seguindo os passos anteriores, o Corpógrafo oferece ferramentas de busca e extração de conhecimento de corpus, como um concordanciador com suporte para pesquisas utilizando expressões regulares, gerador de n-grama (sendo 5 o tamanho máximo possível para o n-grama), extratores de terminologia, relações semânticas e mapas conceituais, dentre outras.

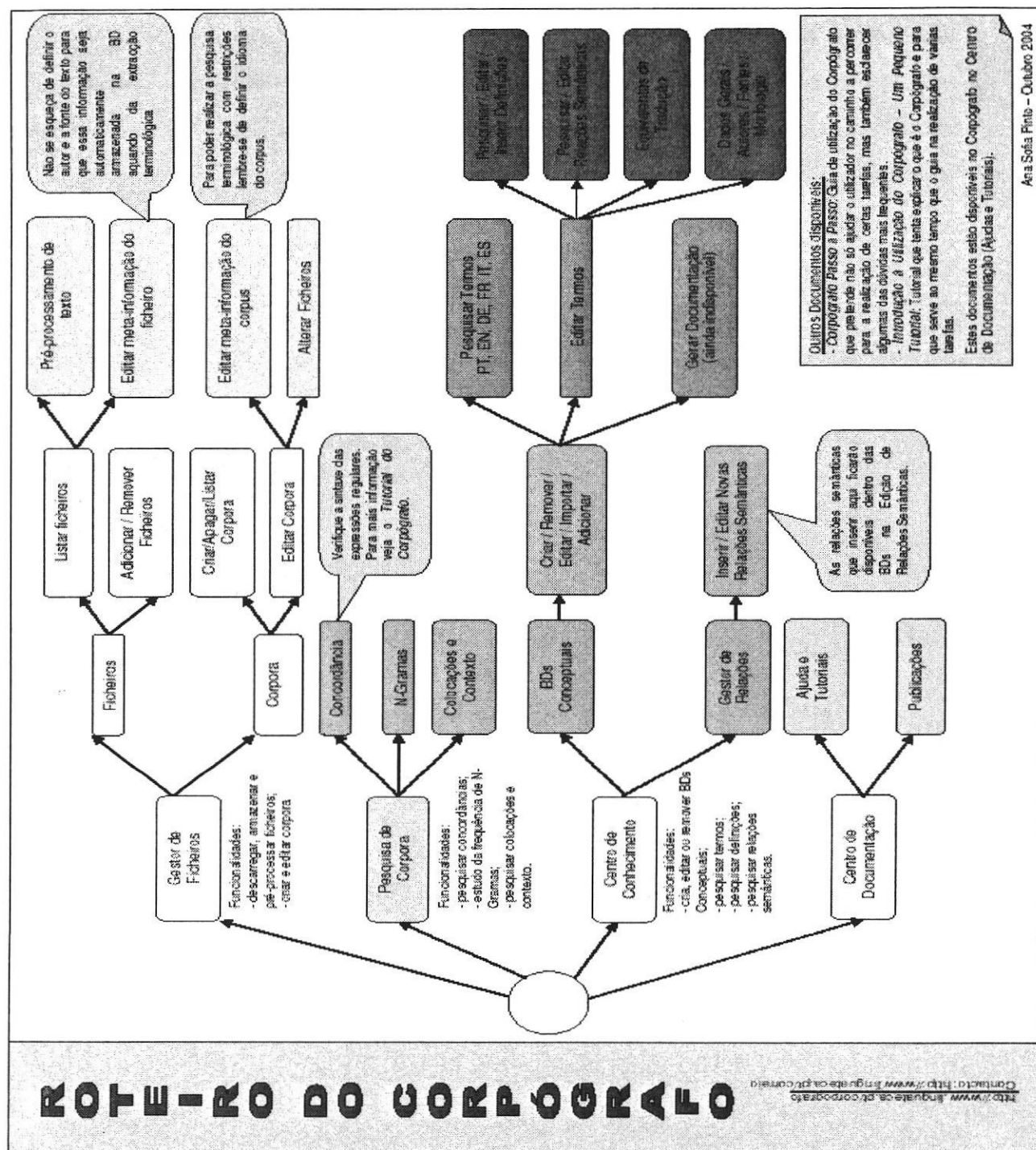
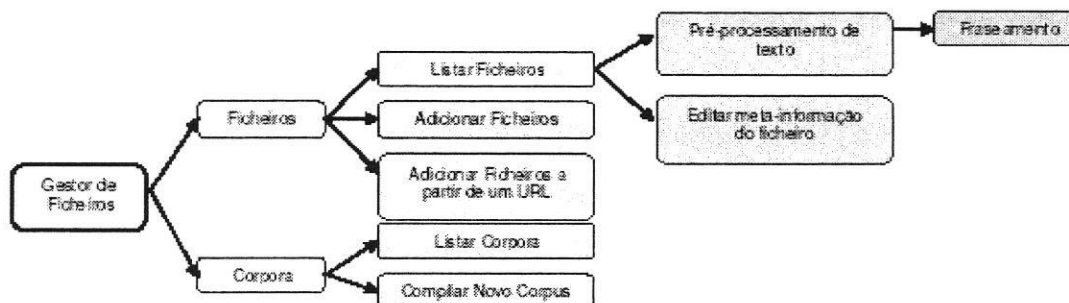


Figura 1: Roteiro completo do Corpógrafo  
 (<http://poloclup.linguatca.pt/corpgrafo/docs/roteiro.pdf>)

## CORPÓGRAFO PASSO A PASSO

**Gestor de Ficheiros:** O *Gestor de Ficheiros* é a base do Corpógrafo. Aqui encontrará as ferramentas necessárias para carregar, armazenar e pré-processar Ficheiros, assim como para criar e preparar os seus corpora de forma a torná-los pesquisáveis.



### Como carregar Ficheiros para o Corpógrafo?

No *Gestor de Ficheiros*, clique em "Adicionar Ficheiros". Em seguida clique em "Browse" para seleccionar o Ficheiro que pretende carregar. Depois de seleccionar o Ficheiro, clique em "Carregar" para que ele seja armazenado no Corpógrafo. Logo que o ficheiro tenha sido carregado, será direccionado para a área de Edição de Meta-dados do ficheiro.

### Como carregar um Ficheiro a partir de um URL?

No *Gestor de Ficheiros*, clique em "Adicionar Ficheiros". Em seguida, no menu "Adicionar Ficheiro" do lado direito, clique em "A partir de um URL". Indique o URL do ficheiro que pretende carregar sem o prefixo `http://`, uma vez que este já se encontra digitado. Por fim clique em "Adicionar" para que o ficheiro seja armazenado no Corpógrafo. Logo que o ficheiro tenha sido carregado, será direccionado para a área de Edição de Meta-dados do ficheiro.

### Ao abrir o Ficheiro, o texto não aparece ou está todo numa só linha.

Muitas vezes o sistema que extrai o texto dos Ficheiros originais não consegue reconhecer quebras de linha. Sempre que isso ocorre, o texto é constituído apenas por uma linha, contendo todas as frases do texto, ou não é visível. Para solucionar este problema, no menu "Operações", clique em "Frasesar". Não se esqueça de gravar!

### Depois de fazer o Frasesamento o texto continua a não aparecer.

Em algumas ocasiões o sistema não consegue extrair texto, muitas vezes devido à qualidade do ficheiro. Por exemplo, há ficheiros PDF que parecem ser de texto, mas que são apenas imagens. Quando isto acontece, o ficheiro em questão não pode ser usado.

### Qual é a utilidade de "Verificar Frasesamento"?

O Corpógrafo faz a divisão automática dos textos em frases, no entanto, podem haver algumas falhas nessa divisão. Ao clicar em "Verificar Frasesamento", pode verificar se a divisão foi ou não bem feita. Note que, depois de fazer o Frasesamento terá de gravar as alterações antes de Verificar o Frasesamento. Para corrigir qualquer falha de Frasesamento, terá de o fazer na interface de edição de texto.

### Como Editar um texto?

Para editar o texto basta clicar em qualquer parte do mesmo. Pode apagar ou acrescentar o que quiser, não esquecendo que, tal como num documento Word, é preciso gravar todas as alterações. Para gravar as alterações basta clicar em "Gravar" no menu "Operações" do lado direito.

### Como gravar um ficheiro para o meu computador?

Neste momento o Corpógrafo não oferece nenhuma funcionalidade que permita gravar um Ficheiro no computador. No entanto, pode seleccionar o texto inteiro (carregue CTRL+A), copiá-lo e colá-lo num documento Word ou NotePad.

### Como criar um Corpus?

No *Gestor de Ficheiros*, clique em "Compilar Novo Corpus". Atribua um nome ao Corpus que deseja criar e preencha os campos que desejar. Em seguida, coloque um visto à frente dos Ficheiros que pretende que façam parte do Corpus. Por fim, clique em "Criar".

### Como apagar um Corpus?

No *Gestor de Ficheiros*, clique em "Listar Corpora". Clique no Corpus que pretende eliminar. Por fim, clique em "Remover Corpus", no menu do lado direito.

**Figura 2: Funcionalidades do Gestor de Ficheiros do Corpógrafo**  
([http://poloclup.linguatca.pt/ferramentas/gc/docs/corp\\_passo\\_passo.pdf](http://poloclup.linguatca.pt/ferramentas/gc/docs/corp_passo_passo.pdf))

### 2.1.2. O ToolKit BootCaT

O BootCaT<sup>11</sup>, extrator automático de *córpus* e de termos (do inglês “Bootstrapping Corpora and Terms”), propõe a montagem de *córpus*, de modo iterativo, a partir de textos obtidos na Web. O BootCaT é composto por várias ferramentas escritas em Perl<sup>12</sup>, que foram projetadas para executar pequenas partes do processo de montagem de *córpus*.

Basicamente, o processo de montagem de *córpus* do BootCaT é composto de quatro passos:

- 1) construir um *córpus* automaticamente a partir de buscas ao Google<sup>13</sup> utilizando um pequeno conjunto de sementes (*seeds*);
- 2) extrair novas sementes desse *córpus*;
- 3) utilizar essas novas sementes para novas buscas ao Google, cujos textos recuperados serão concatenados ao *córpus*, aumentando-o;
- 4) extrair novas sementes desse *córpus* complementado, e assim por diante. A montagem de *córpus* proposta pelo BootCaT segue o diagrama da Figura 3.

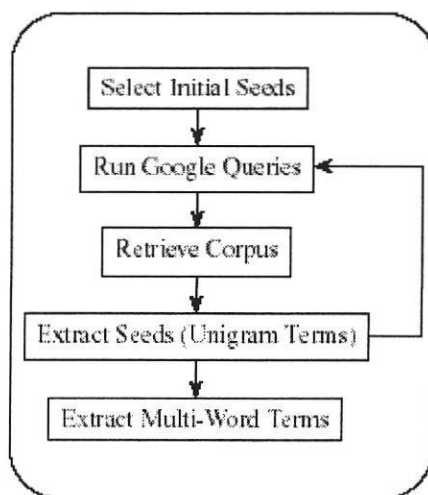


Figura 3: Fluxo de montagem de um *córpus* no BootCaT [4]

O primeiro passo é selecionar as sementes iniciais. Isso é feito manualmente, e boas sementes são termos típicos em textos do domínio específico do qual busca-se construir a

<sup>11</sup> <http://sslmit.unibo.it/~baroni/bootcat.html>

<sup>12</sup> <http://www.perl.com>

<sup>13</sup> <http://www.google.com.br/>

amostragem. No segundo passo, essas sementes são combinadas entre si e algumas dessas combinações (à escolha do usuário) são enviadas como buscas ao Google. No terceiro passo, as URLs retornadas das buscas são processadas para obter-se apenas o texto contido nelas, convertendo-as para texto puro e “limpando-os”, quando for possível. São aproveitados somente os formatos (ou linguagem de marcação) HTML e TXT. Nesse momento, um primeiro corpus já está formado. Desse primeiro corpus são extraídos *unigramas* (termos com apenas uma palavra), e a frequência de cada unigrama obtido no corpus é apurada. Sabendo-se a frequência de cada unigrama, esses podem ser comparados entre si. A relevância de cada unigrama é mensurada utilizando a medida estatística *log odds ratio* [4], com o apoio de um corpus de referência na mesma língua. Uma lista de unigramas, ordenada pela relevância calculada pela medida *log odds ratio*, é então gerada, e os primeiros elementos da lista são considerados bons candidatos a sementes. Caso o corpus obtido até o momento não seja satisfatório (seja pequeno, por exemplo), podemos eleger os primeiros unigramas da lista como novas sementes e repetir o processo, voltando ao segundo passo. Segundo Baroni e Bernardini [4], corpus representativos podem ser montados com poucas sementes iniciais (entre 5 e 15), e também afirmam que com duas ou três iterações é possível obter um corpus satisfatório.

O BootCaT também dispõe de ferramentas para extração de termos com mais de uma palavra, ou termos multi-palavra. Para tal propósito, precisamos de duas listas, ambas obtidas no corpus de referência: uma de *conectores* e uma de *stopwords*. Conectores são compostos por palavras ou bigramas que ocorrem frequentemente entre dois unigramas e *stopwords* são termos muito frequentes, geralmente formados por palavras de classe fechada de uma língua como os artigos, as conjunções, as preposições e os pronomes que não são conectores. As listas descritas acima não precisam necessariamente ser obtidas pelo BootCaT, podem ser dadas ou obtidas de outras fontes. Com as listas acima é possível definir o que são termos multi-palavra, segundo as restrições abaixo:

1. Contém ao menos um unigrama;
2. Não contém *stopwords*;
3. Podem ter conectores, desde que esses não estejam nas extremidades do termo e não sejam consecutivos;
4. Têm frequência maior que um limiar (*threshold*), que é relativo ao tamanho do termo;

5. Não podem ser parte de termos multi-palavras maiores com frequência superior a  $k * fq$ , onde  $k$  é uma constante entre 0 e 1 (normalmente  $k$  é um valor perto de 1) e  $fq$  é a frequência do termo atual;
6. Reciprocamente, não podem conter termos multi-palavras menores com frequência superior a  $(1/k) * fq$ ;

Os termos multi-palavras são procurados recursivamente, inicialmente buscando por bigramas e depois concatenando palavras à esquerda e à direita, na busca de um  $(n+1)$ grama. Parâmetros como a frequência mínima para bigramas (utilizado para calcular o limiar da restrição 4) e o valor de  $k$  das restrições 5 e 6 devem ser informados pelo usuário.

O BootCaT é extremamente modular: para executar o processo de montagem de córpus e extração de termos são utilizadas várias ferramentas, sendo que o resultado de cada ferramenta serve de entrada para outra. Essa característica nos permite utilizar subconjuntos de ferramentas, conferir os arquivos de saída intermediários, adicionar novas ferramentas, substituir uma ferramenta ou alterar uma ferramenta sem preocupar-se com as outras, apenas cuidando para que ela aceite o mesmo tipo de entrada e produza o mesmo tipo de saída. Essa característica reduz re-implementações de algoritmos com implementações consolidadas, evitando a replicação desnecessária de código. Alterações intuitivamente complexas, como adaptações de ferramentas para trabalhar com línguas diferentes, têm sido experimentadas e comprovam os benefícios das ferramentas modulares. Adaptações para o BootCaT foram feitas para construção de córpus em língua japonesa [6], com taxas encorajadoras de reaproveitamento de ferramentas e código. O Apêndice 1 contém uma breve descrição dos scripts do BootCaT que são utilizados na construção de córpus.

### 2.1.3. A API do Google<sup>14</sup>

As buscas e a recuperação das URLs dessas buscas requisitadas pelo BootCaT ao Google são possíveis por meio da API (Interface para Programação de Aplicativos) do Google. Essa API permite ao programador enviar e recuperar facilmente uma busca feita ao Google. Ela contém bibliotecas em várias linguagens (como Perl, Java e Visual Studio.NET) e possibilita ao programador utilizar rotinas de busca diretamente ao Google e obter os resultados em formato fácil de recuperar, como mostrado na Figura 4. Nessa figura, temos vários campos importantes,

---

<sup>14</sup> <http://www.google.com/apis/>

como a URL, o título (Title) e um trecho do conteúdo da URL (snippet). No BootCAT, apenas o campo URL é aproveitado.

```
[
  URL = "http://www.icmc.sc.usp.br/"
  Title = "Instituto de Ciências Matemáticas e de Computação - <b>ICMC</b>-USP"
  Snippet = "O <b>ICMC</b> está localizado no Campus da USP em São Carlos - SP e
oferece cursos nas<br> áreas de Matemática e Ciência da Computação."
  Directory Category = {SE="",
FVN="Top/World/Português/Referência/Educação/Universidades_e_Ensino_Superior/Brasil/Uni
versidade_de_São_Paulo"}
  Directory Title = "<b>ICMC</b> - Instituto de Ciências Matemáticas e de Computação"
  Summary = "O <b>ICMC</b> está localizado no Campus da USP em São Carlos - SP e
oferece cursos nas<br> áreas de Matemática e Ciência da Computação."
  Cached Size = "16k"
  Related information present = true
  Host Name = ""
],
```

**Figura 4: Estrutura de um elemento retornado por uma busca feita pela API do Google**

Para a utilização da API do Google, e conseqüentemente do BootCaT, é necessário obter a licença de uso dessa no site do Google. Para obter essa licença, o usuário precisa cadastrar-se, e a chave da licença lhe é enviada por e-mail. Essa licença permite que o usuário execute diariamente até 1000 buscas e retorne no máximo 10000 resultados.

#### **2.1.4. Extração automática de termos**

Para o projeto de desenvolvimento de uma estrutura conceitual (ontologia) para a área de Nanotecnologia, que ampara o Portal da Nanotecnologia, foi compilado um corpus de língua inglesa da referida área de conhecimento a partir de textos da Web, cuja extensão é de 2.570.792 de palavras. Esse corpus (chamado aqui de Nano) é composto por 22 livros e 200 artigos de revistas, além de 9.982 resumos de artigos, com seus respectivos títulos, e palavras-chaves e lista de tópicos (subjects) quando presentes. Os artigos, em especial, foram compilados dos periódicos (i) *Journal of Nanoscience and Nanotechnology*<sup>15</sup>, (ii) *Nanotechnonology*<sup>16</sup>, (iii) *Nature*<sup>17</sup> e (iv)

<sup>15</sup> <http://www.aspbs.com/html/a0600frm.htm>

*Science*<sup>16</sup>. Os resumos de artigos foram compilados do site Web of Nanotechnonology<sup>19</sup> sobre os tópicos Nanotechnology, Genomics e Nanobiotechnology. Este cópús, criado por um membro da equipe do referido projeto, tem sido usado para o desenvolvimento da ontologia para o Portal da Nanotecnologia pelos três métodos de extração de termos usados no projeto: um método estatístico baseado na frequência e proposto em [15] para extração de uni, bi e trigramas; um método baseado em conceitos de redes complexas; e o método do BootCaT. Os dois primeiros serão comentados na Seção 2.2; o método do BootCaT foi descrito na Seção 2.1.2. A mesma lista de *stopwords* foi usada por todos os métodos.

A ontologia para o Portal ainda está sendo desenvolvida e neste trabalho utilizamos a sua terceira versão, apresentada de forma resumida no Apêndice 2. Esta ontologia está sendo construída com os resultados dos 3 métodos de extração de termos citados acima, e mais outras fontes, como glossários, outras ontologias e o próprio conhecimento da área que um dos membros do projeto possui.

Para a comparação dos métodos foram utilizadas duas métricas: a **precisão**, que é a razão entre as respostas corretas pelo total de respostas obtidas pelo método e a **revocação (recall)**, que é a razão das respostas corretas pelo total de respostas corretas possíveis. O total de respostas corretas possíveis é calculado com as listas de unigramas, bigramas e trigramas da versão 3 da ontologia que vieram do cópús, e, portanto, só faz sentido falar em **revocação** quando a ontologia estiver sendo comparada.

## 2.2. Trabalhos Relacionados

Um dos objetivos deste trabalho foi a avaliação do método de extração de termos do BootCaT. Para esta avaliação, os termos extraídos do cópús Nano pelo BootCaT foram comparados com dois outros métodos de extração: um baseado em estatística, que aceita como termos (uni, bi e trigramas) apenas se estes tiverem uma frequência maior que um limite mínimo (o limite utilizado é 22 para unigramas e está relacionado com o tamanho do cópús Nano); e outro, que modela o cópús como uma rede complexa e usa métricas da rede para a extração de unigramas, somente.

---

<sup>16</sup> <http://www.iop.org/EJ/S/3/451/YfSAujvDHdF2s7uXEE6cyg/journal/Nano>

<sup>17</sup> <http://www.nature.com/cgi-taf/DynaPage.taf?file=/nature/journal/v433/n7025/index.html>

<sup>18</sup> <http://www.sciencemag.org/>

<sup>19</sup> <http://access.isiproducts.com/dwon>

O método baseado em estatística, um dos 15 métodos desenvolvidos em um mestrado do ICMC [5], faz um corte na frequência, ou seja, aceita como termos apenas se estes tiverem uma frequência acima do valor de corte. Para este método foram utilizadas rotinas do pacote NSP (*N-gram Statistic Package*)<sup>20</sup>. O NSP foi implementado por Ted Pedersen, Satanjeev Banerjee e Amruta Purandare da Universidade de Minnesota, Duluth, é escrito em Perl e possui dez medidas estatísticas de associação para bigramas e duas para trigramas, enquanto para unigramas apenas a frequência pode ser medida.

O outro método baseia-se nos conceitos de redes complexas [7, 8]. As redes complexas são grafos basicamente, com a ressalva de que apresentam uma diversidade de técnicas e modelos que auxiliam no entendimento e previsão do comportamento dos sistemas. Três conceitos merecem destaque no estado da arte em redes complexas [7]: as redes *small-world*, o coeficiente de clusterização (*clustering coefficient*) e as redes livres de escala (*scale-free*). O conceito “livre de escala” refere-se ao fato de que, mesmo enormes, a maioria das redes apresenta um caminho relativamente curto entre quaisquer dois nós. Já o coeficiente de clusterização quantifica a tendência de aglomeração dos nós da rede. A maioria das redes complexas apresenta um alto coeficiente de clusterização, quando comparado com o coeficiente de uma rede randômica de mesmo número de nós e arestas. Medidas da rede, como graus de entrada e saída dos nós, pesos das arestas e os conceitos mencionados acima, podem ser utilizadas para extração de termos de um corpus modelado como uma rede complexa.

Esses dois últimos métodos foram aplicados no corpus Nano por dois membros da equipe de trabalho, descrita na Seção 3.2.

### 2.3. Análise Crítica

A construção manual de corpus para experimentos lingüísticos é uma tarefa muito custosa e tem motivado o desenvolvimento de aparatos computacionais para atenuar estes custos, tais como o Corpógrafo e o BootCaT. O Corpógrafo, embora englobe muitas funcionalidades tais como suporte a conversão de vários tipos de texto e organização de corpus, requer que o usuário já tenha recolhido o material que formará o corpus. Além disso, o Corpógrafo é um sistema fechado, o que impede a colaboração de outras pessoas. O BootCaT, entretanto, é totalmente

---

<sup>20</sup> <http://www.d.umn.edu/~tpederse/nsp.html>

aberto e permite que melhorias sejam agregadas, novas ferramentas sejam desenvolvidas e acopladas e até mesmo a substituição e remoção de ferramentas. Por ser tão maleável, a manipulação das ferramentas do BootCaT exige conhecimento avançado de comandos do sistema operacional e de linguagens de programação, que inviabiliza o seu uso por vários interessados na montagem de corpus que nem sempre detêm este tipo de conhecimento.

Neste projeto tentamos produzir algo que atenda os interessados tanto no auxílio computacional para a montagem de corpus quanto no estudo, compreensão, customização e colaboração da ferramenta em si. Além disso, como o BootCaT também oferece métodos para a extração de termos simples e multi-palavras e o projeto do Portal da Nanotecnologia, que se propunha a oferecer uma taxonomia para essa área, precisava de ajuda de métodos automáticos de extração de terminologia. Esse projeto já estava sendo ajudado por outros métodos de extração para auxiliar no levantamento de termos (sendo um método tradicional e outro recente), e então resolvemos também gerar uma lista de candidatos para que estes pudessem ajudar na construção da taxonomia e, além disso, pudéssemos fazer uma comparação mais ampla entre os métodos.

### 3. Estado atual do trabalho

#### 3.1. Projeto

O projeto desenvolvido possui duas partes: 1) utilizar as tecnologias estudadas nesse projeto no **Portal de Nanotecnologia**, especificamente, utilizar o BootCaT para a extração de termos simples e multi-palavras e comparar os termos obtidos com a ontologia e com outros dois métodos (frequência e redes complexas); 2) dar suporte à construção de um módulo do ambiente a ser desenvolvido no projeto **Extração automática de termos e elaboração colaborativa de terminologias para o intercâmbio de conhecimento especializado**, especificamente, acoplar o BootCaT aos módulos já previstos nesse projeto para facilitar a criação do *cópus* de trabalho.

##### 3.1.1. Extração de unigramas e termos multi-palavras e comparação entre métodos

Foram construídas listas de unigramas, bigramas e trigramas usando as ferramentas do BootCaT. Essas listas foram comparadas com outros dois métodos: o de frequência e baseado em conceitos de redes complexas.

No método que usou a frequência como medida estatística, as frequências mínimas requeridas para que os candidatos a termo fossem considerados foram 23, 6 e 6 para unigramas, bigramas e trigramas, respectivamente. Unigramas e bigramas com pelo menos uma *stopword* foram descartados (método OR), enquanto trigramas só foram descartados quando compostos exclusivamente de *stopwords* (método AND).

O método que modela o *cópus* como uma rede complexa obteve três listas de unigramas, a partir de três métricas distintas da rede: grau de entrada sem, peso grau de entrada com peso e clusterização.

Duas formas de comparação foram propostas: avaliação dos 3 métodos pela da intersecção de seus elementos e avaliação com a versão 3 da ontologia, calculando a precisão e a revocação. Os métodos também colaboraram para a criação da versão 4 da ontologia, com os termos recuperados que não estavam presentes na versão 3.

### 3.1.2. Suporte à extração automática de corpus

Com relação ao projeto *Extração automática de termos e elaboração colaborativa de terminologias para o intercâmbio de conhecimento especializado* foram desenvolvidas e adaptadas ferramentas para serem incluídas no ambiente Web denominado **e-Termos**, como um módulo extra aos 5 previstos e mostrados no Anexo 1 que traz um esquema genérico da arquitetura computacional do Ambiente.

O Ambiente Web Colaborativo proposto no projeto de doutorado é, sob uma perspectiva genérica, um sistema Web cuja entrada principal é um corpus-alvo de um determinado domínio do conhecimento, e a saída um produto terminológico (glossário, dicionário etc.) do domínio em questão. Para facilitar o uso de tal ambiente por usuários que não tem familiaridade com a construção de corpus, o projeto de graduação projetou o módulo de construção automática de corpus a partir de um conjunto pequeno de termos de uma área e um outro membro da equipe implementou a interface gráfica que chamam as rotinas do BootCaT, que são de domínio público. Além de incorporar estas rotinas, também apontamos suas limitações, para que possamos alterá-las ou acrescentar novas ferramentas para minimizar essas restrições.

## 3.2. Equipe

Este projeto de graduação está inserido no escopo de dois projetos maiores. O primeiro projeto, *Extração automática de termos e elaboração colaborativa de terminologias para o intercâmbio de conhecimento especializado*, é coordenado pela Profa. Dra. Gladis Maria de Barcellos Almeida do Departamento de Letras (DL) da Universidade Federal de São Carlos (UFSCar) e pela Profa Sandra Maria Aluísio do ICMC-USP. Este projeto de graduação adaptou as ferramentas do BootCaT para suporte à montagem de corpus em um ambiente Web. Marcos Felipe Tonelli de Carvalho, aluno do Bacharelado em Informática e bolsista do NILC, desenvolveu uma interface para as ferramentas serem acessadas no ambiente Web. No segundo projeto, Desenvolvimento de uma estrutura conceitual (ontologia) para a área de Nanotecnologia, que é coordenado pela Profa. Sandra Maria Aluísio, temos também os professores Osvaldo Novais de Oliveira Jr. do IFSC-USP (especialista do domínio), Maria das Graças Volpes Nunes do ICMC-USP e Gladis Maria de Barcellos Almeida do DL da UFSCar; os alunos de doutorado Ariani Di Felippo e Leandro Henrique Mendonça de Oliveira que trabalham no NILC, e o aluno

de graduação, também desenvolvendo um projeto de graduação orientado pela Profa. Maria das Graças Volpes Nunes, Lucas Antiqueira.

### **3.3. Descrição das Atividades Realizadas**

As atividades pontuais realizadas são itemizadas abaixo.

#### **3.3.1. Avaliação dos scripts do BootCaT: levantando suas limitações**

O BootCaT foi instalado em um computador do NILC e um experimento foi realizado para compor um corpus, sem fazer iterações nem extrair termos multi-palavras. Os tipos de arquivo endereçados pelas URLs retornadas pelo Google estavam distribuídos da seguinte maneira: 556 hipertextos, 46 PDF, 6 docs, 5 TXT, 1 PS, nenhum RTF e 5 de outros tipos (como PPT), totalizando 619 URLs. Observamos então que cerca de 10% das URLs retornadas pelo Google foram descartadas pela ferramenta de obtenção de textos da Web, entre documentos PDF, DOC, PS e RTF. Nenhum conteúdo de páginas com frames e redirecionamentos foi obtido pela ferramenta.

As ferramentas do BootCaT que foram utilizadas para a geração do primeiro corpus foram profundamente estudadas, que resultou num roteiro de uso do BootCaT, desde a sua instalação até a construção do primeiro corpus e extração dos unigramas. Esse roteiro, juntamente com uma breve descrição do BootCaT, foi apresentado aos alunos da disciplina de pós-graduação *SCE-5819-2: Tópicos em Inteligência Artificial*. Foi proposto aos alunos que seguissem o roteiro apresentado para a construção de um corpus e a extração de unigramas. As dificuldades relatadas pelos alunos contribuíram consideravelmente para a avaliação do BootCaT, desde falha de conversão como problemas de portabilidade.

#### **3.3.2. Inclusão de novas ferramentas para conversão de tipos de arquivo**

Uma das grandes limitações do BootCaT são suas restrições de tipos de arquivo que são convertidos e incluídos ao corpus. As URLs retornadas pelo Google que se referem a arquivos em formato PDF (Portable Document Format) e DOC (Microsoft Word Document), por exemplo, são ignorados pela ferramenta. Normalmente, esses arquivos contêm textos mais informativos e confiáveis que os em formato HTML. Buscando ampliar a gama de tipos de arquivos a serem aproveitados, alguns programas, todos eles de livre distribuição e disponíveis na Web, foram

estudados e dois deles se mostraram eficientes ao propósito de transformar os tipos de arquivo citados acima para texto, e foram incorporados à ferramenta do BootCaT que obtêm os textos a partir das URLs.

O aplicativo **xpdf**<sup>21</sup>, dentre outras funções, converte arquivos PDF para texto. A grande maioria dos arquivos PDF testados foi corretamente convertida para texto, embora arquivos PDF protegidos por criptografia não possam ser convertidos. Além disso, existem versões do xpdf tanto para sistemas Windows quanto para Linux.

O aplicativo **wvware**<sup>22</sup> permite converter arquivos do tipo DOC para texto. Nos testes realizados, este aplicativo fez boas conversões, sem deixar resíduos de conversão. Assim como o xpdf, o wvware também tem versões tanto pra sistema Linux quanto Windows.

### **3.3.3. Extração de unigramas usando o BootCaT**

Primeiramente *tokenizamos* o *cópus*, que consiste em separar cada palavra do *cópus*, e geramos um arquivo de *cópus tokenizado* com uma palavra por linha. Este *cópus* também foi transformado para minúsculo, para não discriminar maiúsculas de minúsculas. Com o *cópus tokenizado*, calculamos a frequência de cada palavra distinta do *cópus*, no qual cada linha contém a palavra seguida da sua frequência no *cópus*, ordenada decrescentemente pela frequência.

O próximo passo foi calcular a medida estatística *log odd ratio* para cada palavra da lista de frequência. Nesta tarefa, precisamos da lista de frequência de um *cópus* de referência, e escolhemos o *cópus* Brown<sup>23</sup>. Para calcularmos a medida *log odd ratio*, precisamos de um arquivo com a frequência de cada palavra do *cópus* ao lado da frequência dessa palavra no *cópus* de referência +1 (mais um). Calculamos o *log odd ratio* de cada palavra e selecionamos os 10% melhores unigramas classificados pela medida, gerando uma lista com 7343 unigramas. Dessa lista, foram removidos os unigramas que também estavam na lista de *stopwords*, resultando numa lista final com 7297 unigramas. Os comandos relativos aos passos citados acima para a geração de unigramas podem ser analisados no Apêndice 3.

### **3.3.4. Extração de bigramas e trigramas usando o BootCaT**

---

<sup>21</sup> [www.foolabs.com/xpdf/](http://www.foolabs.com/xpdf/)

<sup>22</sup> <http://wvware.sourceforge.net/>

<sup>23</sup> [clwww.essex.ac.uk/w3c/corpus\\_ling/content/corpora/list/private/brown/brown.html](http://clwww.essex.ac.uk/w3c/corpus_ling/content/corpora/list/private/brown/brown.html)

Para este passo, precisamos da lista de unigramas obtida na Seção 3.3.3. Então extraímos conectores de uma e duas palavras, e escolhemos os 10% primeiros de uma palavra e os 5% primeiros de duas palavras. Poderíamos extrair a lista de *stopwords* da lista de frequência do *corpus* Brown, mas usamos a lista de *stopwords* do Apêndice 4, que também foi utilizada pelos dois outros métodos de extração descritos na Seção 3.1.1. Como vimos na Seção 2.1.2, a extração de termos multi-palavras começa com a extração de bigramas. A partir dos bigramas é que podemos extrair trigramas olhando no unigrama que sucede ou precede o bigrama, e assim por diante. Foram extraídos 6,5% dos possíveis bigramas contidos no *corpus* Nano, para que o número de bigramas retornados se equiparasse com o número extraído pelo método da frequência. Com esses bigramas, partimos para a extração de trigramas, que consiste em tentar eleger bigramas que são partes de trigramas. A extração de trigramas baseia-se nas regras de formação de termos multi-palavras, descritas na seção 2.1.2. Da lista dos 6,5% bigramas de maior frequência escolhidos, a menor frequência é 5. O valor que escolhemos para  $k$  foi de 0.75, o mesmo usado no trabalho dos autores do BootCaT, e para um trigrama  $X$  que contém o bigrama  $Y$  ser classificado com trigrama, a frequência de  $X$  de ser superior a  $k*(\text{frequência de } Y)$ . A Tabela 1 mostra alguns bigramas, e suas respectivas frequências no *corpus*, que estão contidos nos trigramas da Tabela 2. Os termos em negrito são os termos que estão nas listas finais de bigramas e trigramas obtidas pelo BootCaT.

| Bigramas              | Frequência |
|-----------------------|------------|
| biochemical research  | 495        |
| research methods      | 494        |
| <b>scanning probe</b> | 123        |

Tabela 1: Candidatos a bigramas

| Trigrama                            | Frequência |
|-------------------------------------|------------|
| scanning probe microscopy           | 51         |
| scanning probe microscope           | 23         |
| scanning probe microscopes          | 15         |
| <b>Biochemical research methods</b> | 490        |

Tabela 2: Candidatos a trigramas

O trigrama “*scanning probe microscopy*” não foi incluído na lista de trigramas, embora seja um termo da área, pois sua frequência é menor que a mínima exigida:

$$\text{Frequência mínima} = k*(\text{frequência de “scanning probe”}) = 0.75 * 123 \approx 93$$

Se o corpus fosse maior ou se sofresse um processo de singularização a frequência de “*scanning probe microscopy*” seria maior e ele seria escolhido.

Já o trigramma “*biochemical research methods*” foi incluído na lista final de trigramas, excluindo portanto os bigramas “*biochemical research*” “*research methods*” da lista de bigramas, pois sua frequência é maior que a frequência mínima:

$$\text{Frequência mínima} = k * (\text{frequência de “biochemical research”}) = 0.75 * 495 \approx 372$$

A frequência mínima exigida para trigramas é a frequência do menor bigrama (5) vezes  $k$  (0.75), que é 3.75, e como frequência é um valor inteiro, o mínimo exigido foi 4.

Os comandos relativos aos passos para a obtenção de lista de bigramas e trigramas podem ser vistos no Apêndice 3.

### 3.4. Resultados Obtidos

#### 3.4.1. Extração de unigramas e termos multi-palavras

Na Tabela 3 são apresentados os números de uni, bi e trigramas da Ontologia, extraídos pelo BootCaT e pelo método da frequência, respectivamente nas colunas 2, 3 e 4.

| N-grama   | Ontologia | BootCaT | Frequência |
|-----------|-----------|---------|------------|
| Unigramas | 1187      | 7297    | 7645       |
| Bigramas  | 2261      | 6780    | 1954       |
| Trigramas | 2530      | 152     | 3216       |

Tabela 3: Número dos candidatos a termos obtidos pelos métodos do BootCaT, Frequência e da ontologia

O método baseado em conceitos de redes complexas extraiu apenas unigramas, mas extraiu três listas, utilizando três métricas da rede (Tabela 4).

| Métrica da rede               | Quantidade de unigramas |
|-------------------------------|-------------------------|
| Grau de entrada sem peso      | 3107                    |
| Grau de entrada com peso      | 2350                    |
| Coefficiente de clusterização | 3391                    |

Tabela 4: Número de candidatos a termos obtido das redes complexas

| N-grama   | Termos na Intersecção | Precisão | Revocação |
|-----------|-----------------------|----------|-----------|
| Unigramas | 651                   | 8.92%    | 54.84%    |
| Bigramas  | 248                   | 3.65%    | 10.96%    |
| Trigramas | 9                     | 5.92%    | 0.35%     |

**Tabela 5: Comparação entre os termos obtidos utilizando o BootCaT e a ontologia**

A Tabela 5 apresenta o resultado da comparação entre a ontologia e os termos obtidos pelo BootCaT. O método da frequência teve mais sucesso que os outros dois métodos como podemos ver na intersecção dos termos com a ontologia. Porém, os unigramas, bigramas e trigramas retornados pelo BootCaT que não estão na ontologia servirão para alimentar a quarta versão desta, se após o julgamento por um especialista forem considerados termos da área. Na Tabela 6 temos o resultado da comparação entre a ontologia e os termos obtidos pelo método da frequência.

| N-grama   | Termos na Intersecção | Precisão | Revocação |
|-----------|-----------------------|----------|-----------|
| Unigramas | 815                   | 10.66%   | 68.66%    |
| Bigramas  | 252                   | 12.89%   | 11.14%    |
| Trigramas | 30                    | 0.93%    | 1.18%     |

**Tabela 6: Comparação entre os termos obtidos utilizando a frequência e a ontologia**

Na tabela 7, temos a comparação entre os unigramas da ontologia e os unigramas obtidos pela rede complexa.

| Métrica da rede               | Termos na Intersecção | Precisão | Revocação |
|-------------------------------|-----------------------|----------|-----------|
| Grau de entrada sem peso      | 542                   | 17.44%   | 45.66%    |
| Grau de entrada com peso      | 469                   | 19.95%   | 39.51%    |
| Coefficiente de clusterização | 27                    | 0.79%    | 2.27%     |

**Tabela 7: Comparação entre os unigramas obtidos utilizando diferentes métricas da redes complexas e a ontologia**

A comparação entre os métodos mostrada nas Tabelas 8 e 9 visa avaliar a **similaridade** entre esses no ranqueamento de termos. Para que a comparação fosse justa, listas de n-gramas de tamanhos diferentes foram reduzidas ao menor tamanho entre as duas.

| N-grama   | Termos Comparados | Termos na Intersecção | Porcentagem |
|-----------|-------------------|-----------------------|-------------|
| Unigramas | 7297              | 3742                  | 51.28%      |
| Bigramas  | 1954              | 1212                  | 62.02%      |
| Trigramas | 152               | 1                     | 0.65%       |

**Tabela 8: Comparação entre os termos obtidos utilizando o BootCaT e os obtidos pelo corte na frequência**

| <b>Métrica da rede</b>        | <b>Termos Comparados</b> | <b>Termos na Intersecção</b> | <b>Porcentagem</b> |
|-------------------------------|--------------------------|------------------------------|--------------------|
| Grau de entrada sem peso      | 3107                     | 1052                         | 33.85%             |
| Grau de entrada com peso      | 2350                     | 720                          | 30.63%             |
| Coefficiente de clusterização | 3391                     | 23                           | 0.67%              |

**Tabela 9: Comparação entre os unigramas obtidos utilizando diferentes métricas da redes complexas e os unigramas obtidos pelo BootCaT**

### **3.4.2. Suporte à montagem de córpus**

As ferramentas do BootCaT foram preparadas para que fossem utilizadas no ambiente Web do **e-Termos**, e já estão sendo utilizadas a partir de uma interface gráfica, que pode ser vista no Anexo 2. Esse anexo mostra todos os passos necessários para a construção de córpus descartáveis para pesquisa terminológica.

### **3.5. Dificuldades e Limitações**

A busca por ferramentas de conversão é uma tarefa exaustiva, custosa e muitas vezes improdutiva. A maioria das ferramentas de conversão é comercial, o que inviabiliza a inclusão dessas no projeto. Outras não são comerciais, mas demandam interação com o usuário, inviabilizando a automatização do processo.

Segundo a documentação do BootCaT, as ferramentas foram testadas em sistemas tipo Unix, mas não haveria problema em rodar o BootCaT sobre sistemas emulados, como o CygWin. Algumas horas de projeto foram dedicadas à adaptação do BootCaT ao CygWin, devido a entraves devido às diferenças sutis entre os sistemas.

## 4. Conclusões e Trabalhos Futuros

Neste projeto foi realizado o estudo e a adaptação das ferramentas do BootCaT, um conjunto de programas Perl que auxilia na montagem de corpúsculo de textos providos da Web, com o intuito de aprimorar suas ferramentas e prepará-las para que pudessem comportar-se como um módulo de montagem automática de corpúsculo para o projeto *Extração automática de termos e elaboração colaborativa de terminologias para o intercâmbio de conhecimento especializado*. Além disso, suas ferramentas fornecem um método alternativo de extração de termos multi-palavras, para auxiliar na construção da ontologia da área de nanotecnologia, que integrará o *Portal da Rede de Nanotecnologia da USP*.

As ferramentas do BootCaT foram preparadas para que pudessem ser facilmente utilizadas em um ambiente Web. Listas de unigramas, bigramas e trigramas foram extraídas do corpúsculo compilado para o projeto de nanotecnologia, e os candidatos a termos que não pertenciam à ontologia foram enviados para especialistas trabalharem na construção da próxima versão da ontologia.

O suporte à montagem de corpúsculo, incorporado no Ambiente Web, sobrecarrega muito o servidor, pois todo o *download* do corpúsculo, que pode ser de alguns MBytes mesmo para uma pequena lista de termos (*seeds*), fica a cargo do servidor. As buscas são feitas por meio da API do Google, que restringe o número de resultados retornados e obriga a quem for usar esta ferramenta do BootCaT a ter uma senha para utilizar a API do Google, embora esta tenha apenas que ser solicitada. Também existem problemas quando a URL não é bem definida, ou seja, não fica explícito na URL qual é o tipo de arquivo endereçado por ela. Neste caso, a ferramenta do BootCaT tenta, por exemplo, interpretar um PDF como um hipertexto, gerando vários kbytes de puro lixo. Tentamos resolver o problema por meio da API do Google, mas os resultados retornados pelo Google não apresentam o tipo de arquivo que está apontado pela URL.

A conversão entre formatos de arquivos também pode ser melhorada, adicionando novos formatos (como PS e RTF), incorporando uma ferramenta que trata páginas HTML com frames e redirecionamentos e tratando o lixo gerado no processo. Também seria interessante avaliar a qualidade do corpúsculo montado com o processo iterativo do BootCaT, talvez usando-o para gerar um corpúsculo com aproximadamente o mesmo número de palavras do usado neste projeto, e aplicando os mesmos métodos e extração para verificar a intersecção entre os termos extraídos do

cópus compilado manualmente e o cópus obtido pelas ferramentas do BootCaT. Existe uma ferramenta escrita em Java por este aluno que extrai texto de páginas com frames e redirecionamentos com a ajuda do navegador de modo texto **lynx**<sup>24</sup>, que ainda precisa ser incorporada ao BootCaT.

---

<sup>24</sup> <http://lynx.browser.org/>

## Referências Bibliográficas

- [1] GARSIDE, R.; LEECH, G.; MCENERY, A.M. (eds.) (1997). *Corpus Annotation*. Longman.
- [2] LUÍS SARMENTO, BELINDA MAIA & DIANA SANTOS. The Corpógrafo: a Web-based environment for corpora research Proceedings of LREC 2004 . Lisboa, Portugal, 25 May 2004.
- [3] BELINDA MAIA & LUÍS SARMENTO. CG - An integrated Environment for Corpus Linguistics. Poster apresentado na conferência CL2003: CORPUS LINGUISTICS 2003 - Lancaster University (UK), 28 - 31 March 2003.
- [4] M. BARONI e S. BERNARDINI. 2004. BootCaT: Bootstrapping corpora and terms from the web. Proceedings of LREC 2004.
- [5] TELINE, M. F. (2004). Avaliação de métodos para extração automática de terminologia de textos em português. ICMC-USP, São Carlos, 2004. Dissertação de Mestrado.
- [6] M. BARONI e M. UEYAMA. 2004. Retrieving Japanese specialized terms and corpora from the World Wide Web. Proceedings of KONVENS 2004.
- [7] ALBERT, R.; BARABÁSI, A.L. Statistical Mechanics of Complex Networks. *Rev. Modern Phys.*, 74, pp. 47-97, 2002.
- [8] NEWMAN, M.E.J. The Structure and Function of Complex Networks. *SIAM Review* 45, 167-256, 2003.

## Apêndice 1: Resumo das ferramentas de obtenção de corpus do BootCaT

| build_random_tuples.pl |                                             |                                   |
|------------------------|---------------------------------------------|-----------------------------------|
| Descrição              | Gera uma combinação entre sementes          |                                   |
| Sintaxe                | perl build_random_tuples.pl -nX -lY Entrada |                                   |
| Opções                 | -n                                          | X é o tamanho das tuplas geradas  |
|                        | -l                                          | Y é o número de tuplas retornadas |
| Entrada                | Arquivo com as sementes, uma por linha      |                                   |
| Saída                  | As combinações na STDOUT, uma por linha     |                                   |

| collect_urls_from_google.pl |                                                                                                              |                                                   |
|-----------------------------|--------------------------------------------------------------------------------------------------------------|---------------------------------------------------|
| Descrição                   | Faz uma busca ao Google e obtém os links retornados                                                          |                                                   |
| Sintaxe                     | perl collect_urls_from_google.pl -k chave -l lang -c nr Entrada                                              |                                                   |
| Opções                      | -k                                                                                                           | chave é a GOOGLE_API_KEY (obrigatório)            |
|                             | -l                                                                                                           | lang é a língua das páginas buscadas              |
|                             | -c                                                                                                           | nr é número de resultados coletados em cada busca |
| Entrada                     | Arquivo com as buscas, uma busca por linha                                                                   |                                                   |
| Saída                       | Para cada busca, uma linha "CURRENT SEED" seguido da busca e até nr primeiras URLs retornadas, uma por linha |                                                   |

| print_pages_from_url_list.pl |                                                                                                                                                                                      |
|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Descrição                    | Obtém o corpus da Web e o transforma para txt                                                                                                                                        |
| Sintaxe                      | perl print_pages_from_url_list.pl Entrada                                                                                                                                            |
| Entrada                      | O arquivo com URLs que devem ser baixadas e passadas para txt, uma por linha. Apenas as linhas que começarem com "http://" serão processadas. Não há tratamento para links repetidos |
| Saída                        | Para cada URL, uma linha "CURRENT URL" seguido da URL e abaixo o texto obtido na URL                                                                                                 |

| asasas basic_tokenizer.pl |                                                             |
|---------------------------|-------------------------------------------------------------|
| <b>Descrição</b>          | Tokenizador simples<br>Sadasdasdsadsa                       |
| <b>Sintaxe</b>            | perl basic_tokenizer.pl -i -e -a Entrada<br>-i              |
| <b>Opções</b>             | -i transforma todas as letras em minúsculas                 |
|                           | -e informa que é para tratar apóstrofes para língua Inglesa |
|                           | -a descarta caracteres não alfabéticos                      |
| <b>Entrada</b>            | Arquivo do corpus em txt                                    |
| <b>Saída</b>              | O corpus tokenizado, um token por linha                     |

## Apêndice 2: Recorte da Ontologia sobre Nanociência e Nanotecnologia

A ontologia possui seus grandes conceitos (enumerados abaixo) e mais duas entradas chamadas Major Topics Related to N&N e Key Concepts in N&N, que trazem os tópicos mais importantes e as palavras-chaves da área, respectivamente.

### 1. Synthesis, Processing and Fabrication

- Chemical synthesis
- Sol-gel methods
  - Colloidal sol-gel
  - Inorganic polymeric gel
  - Organic polymeric gel
- Molecular beam epitaxy
  - Homoepitaxy
  - Heteroepitaxy
- Laser cooling and trapping
- Self-assembly
  - Chemisorption
  - Physical adsorption (layer-by-layer technique)
- Vapor processing
  - Chemical vapor deposition
  - Physical vapor deposition
- Atom Optics
  - Laser focused atom deposition
  - Laser collimation
  - Fabrication via reactive-ion etching laser focusing
- Nanoscale crystal growth

...

### 2. Materials

- Metals and alloys
  - Aluminum
    - Aluminium alloys
  - Silver
  - Calcium
  - Cobalt

- Copper
  - Copper alloys
- Gold
  - Gold-palladium alloy
- Iron
- Lead
- Lithium
- Magnesium
  - Magnesium alloys
- Magnetite
- Mercury
- Molybdenum
- Nickel
  - Nickel alloy
  - Nickel-iron alloy
- Niobium
- Palladium
- Platinum
- Steel
- Superalloys
- Superconductors
- Titanium
  - Titanium alloy
- Tungsten
- Zinc
  - Zinc alloys

Gases and vapors

...

### **3. Properties and Characterization techniques**

Electrical Properties

- Electronic transport, etc.

Optical Properties

- Photorefractive

- Near-field optics

Linear and nonlinear spectroscopy

- Vibrational spectroscopy
  - Fourier-transformed infrared (FTIR)
  - Correlation spectroscopy
- Raman
- Circular dichroism
- UV-visible spectroscopy
- NMR spectroscopy
- Tribology

...

#### **4. Machines and Devices**

- Molecular and supramolecular nanomachines
- Functional nanostructures incorporating responsive modules
- Photodetectors
  - Quantum well infrared photodetectors
- Sensors and actuators
  - Atomic and molecular sensors
  - Biosensors
  - Gas sensors
  - Chemical sensors
  - Taste sensors
  - Protein-coupled receptors
- Nanomanipulators

...

#### **5. Theories and Computational methods**

- Quantization and confinement phenomena
- Spintronics
- Quantum information theory
- Atomistic simulation methods
- Nanoscale fluid mechanics
- Nanoelectronics
- Molecular dynamics simulation
- High throughput screening
- Combinatory chemistry
- Neural networks

...

## 6. Applications

Acoustics

Telecommunications

Radio

Telephone

Television

Ultrasound

Sound absorber

Sound amplifier

Noise barriers

Radio Frequency Identification Tags

Agriculture

Agrochemicals

Herbicides

Pest Repellents

Pesticides

Airplanes

...

## Apêndice 3: Geração das listas de unigramas, bigramas e trigramas

Os roteiros abaixo foram rodados no shell bash de sistemas do tipo Linux. O Shell de emuladores de Linux (como o CygWin para MS Windows) também deve aceitar esses comandos sem maiores problemas.

### Geração de unigramas

Dado um *cópus* armazenado no arquivo “corpus.txt”, veremos os passos que foram seguidos na extração da lista de unigramas:

O primeiro passo é tokenizar o *cópus*, usando um tokenizador qualquer. A opção “-i” passa o *cópus* todo para minúsculo.

```
./tokenizer2.sh -i corpus.txt > corpusTok.txt
```

Agora já podemos calcular a frequência do *cópus*.

```
cat corpusTok.txt | sort | uniq -c | awk '{print $2,$1}' >
freqCorpus.txt
```

O arquivo “freqCorpus.txt” contém uma lista das palavras do *cópus* ordenada, com a frequência de cada uma delas na frente. Os passos abaixo calculam a medida *log odd ratio* usando a lista de frequência já calculada e a lista de frequência de um *cópus* de referência, todo em minúsculo (pois nosso *cópus* também está em minúsculo). Neste projeto, usamos o *cópus* Brown, denominado “ci\_brown\_tok\_fqs.txt” nos comandos abaixo:

```
nwc=`wc corpus.txt | tr -s " " " " | cut -d" " -f3`
nw2=`perl add1_smoothing2.pl -t ci_brown_tok_fqs.txt freqCorpus.txt
smoothCorpus.add1`
```

```
paste freqCorpus.txt smoothCorpus.add1 | awk '{print $1,$2,$4}' | \
perl log_odds_ratio.pl $nwc $nw2 - | sort -nrk2 > logOdds.txt
```

O comando abaixo ordena os unigramas de acordo com seus valores de *log odd ratio*, e armazena esta lista ordenada em “corpusRank.txt”

```
perl print_rank.pl -f2 logOdds.txt | awk '{print $2,$1}' | sort -nk2 >
corpusRank.txt
```

Abaixo escolhemos os 10% melhores classificados pela medida *log odd ratio* como nossos unigramas.

```
perl get_top_percentage.pl 10 corpusRank.txt | awk '{print $1}' >
candUnigramList.txt
```

Agora só falta remover os unigramas dessa lista que são *stopwords*, e gerar a lista final de unigramas em unigramList.txt

```
perl simple_filter.pl -s stopwordsList.txt candUnigramList.txt >
unigramList.txt
```

## Gerando bigramas e trigramas

Para gerarmos bigramas e trigramas, precisaremos dos arquivos com a lista de conectores (unigramList.txt) e o cópulo tokenizado e transformado para minúsculo (corpusTok.txt), já obtidos na extração de unigramas.

```
./find_conectores.sh uni unigramList.txt corpusTok.txt 10 > uni_con.txt
./find_conectores.sh bi unigramList.txt corpusTok.txt 5 > bi_con.txt
```

Acima, obtemos a lista de conectores, com 10% para conectores com um token e 5% para conectores com 2 tokens.

O comando abaixo conecta conectores com dois tokens para poderem ser tratados com uma palavra só.

```
perl connect_bi_connectors.pl bi_con.txt corpusTok.txt > corpusBitok.txt
```

Agora começamos a extração de bigramas, e pegaremos os 6,5% mais frequentes.

```
./extract_bigram.sh corpusBitok.txt stopwordsList.txt unigramList.txt >
allgrams.txt
awk '(NF==3){print}' allgrams.txt | sort -nrk3 | perl
get_top_percentage.pl 6.5 - > topBigrams.txt
```

Agora vemos se alguns dos bigramas são trigramas na verdade. Informamos o valor de  $k$  (0.75), e geramos uma lista com os termos multi-palavras em “biEtri\_grams.txt”.

```
./extract_multi.sh uni_con.txt bi_con.txt allgrams.txt topBigrams.txt
0.75 > biEtri_grams.txt
```

Tanto os bigramas quanto os trigramas estão nesta lista. Agora é só separar os que têm duas palavras dos que têm três palavras.

```
cat biEtri_grams.txt | awk '{if (NF==2){print}}' > bigramList.txt
cat biEtri_grams.txt | awk '{if (NF==3){print}}' > trigramList.txt
```

Assim temos os bigramas em “bigramList.txt” e os trigramas em “trigramList.txt”.

## Scripts extras

Nos comandos para extração de unigramas, bigramas e trigramas vistos neste apêndice, vimos alguns comandos com extensão “.sh”, que foram criados neste projeto para clarificar a finalidade de alguns comandos realmente obscuros. Abaixo temos o código de cada um destes comandos.

### Tokenizer2.sh

```
#!/bin/sh

help () {
    echo "--> Ferramenta:   Tokenizador"
    echo "    Modo de uso:   $0 [-ih] <arq_corpus>"
    echo "    Dependecies:   -"
    echo "    Resultado:     Arquivo tokenizado em STDOUT"
    exit
}

corpus=$1

while getopts "ih" op; do
    case "$op" in
        h )help;;
        i )echo "Ignorar case insensitive";
            ignoreCase=true;
            corpus=$2;;
    esac
done

if ! [ $corpus ]; then
    help
fi

if [ $ignoreCase = true ]; then
    perl -ne 'if(/CURRENT URL/){print "CURR_DOC\n";next} \
s/\x92/\x27/g; s/^(.)/ _ $1/; s/\s\s/ _ /g; \
s/\-\/\-\/\- \-/g; s/[^\s+]+\.[^\s+]/ _ /g; \
s/[^a-zA-Z\x27\x27s0-9\-]+/_STOP_/g; \
s/([a-z])([A-Z])/$1 $2/g; s/\s+/\n/g; \
print' $corpus |tr 'A-Z' 'a-z' | egrep "[A-Za-z]" |\
perl -ne 'BEGIN{$seen=0} if ( /^CURR_DOC/ ) {print; next} if ( /^_STOP_/ )
{if ($seen==0){$seen=1; print}next;} $seen=0; s/^[x27\-' ]+//g; \
s/[x27\-' ]+$//g; if (/[A-Za-z]/){print}'
else
    perl -ne 'if(/CURRENT URL/){print "CURR_DOC\n";next} \
s/\x92/\x27/g; s/^(.)/ _ $1/; s/\s\s/ _ /g; \
s/\-\/\-\/\- \-/g; s/[^\s+]+\.[^\s+]/ _ /g; \
s/[^a-zA-Z\x27\x27s0-9\-]+/_STOP_/g; \
s/([a-z])([A-Z])/$1 $2/g; s/\s+/\n/g; \
print' $corpus | egrep "[A-Za-z]" |\
perl -ne 'BEGIN{$seen=0} if ( /^CURR_DOC/ ) {print; next} if ( /^_STOP_/ )
{if ($seen==0){$seen=1; print}next;} $seen=0; s/^[x27\-' ]+//g; \
s/[x27\-' ]+$//g; if (/[A-Za-z]/){print}'
fi
```

### find\_conectores.sh

```
#!/bin/sh

help () {
    echo "--> Ferramenta:      Gera lista de conectores"
    echo "      Modo de uso:      $0 <\\"uni|bi\\"> <arq_candidatos_unitermos>
    <arq_corpus_tokenizado> <percentagem>"
    echo "      Dependecies(2):  get_connector_grams.pl; get_top_percentage.pl"
    echo "      Resultado:      a lista de unigramas ou bigramas conectores
    coletados no corpus"
}

if ! ( [ $1 ] && [ $2 ] && [ $3 ] && [ $4 ] ); then
    help
    exit
fi

candsUni=$2
corpusTok=$3
percent=$4

if [ $1 = "uni" ]; then
    perl get_connector_grams.pl 3 $candsUni $corpusTok | \
        grep -v _ | sort | uniq -c | \
        perl -ane 'if ($F[0]>2){print $F[2];print "\\n";}' | sort | uniq -c | \
        awk '{print $2,$1}' | sort -nrk2 | perl get_top_percentage.pl
    $percent - | \
        awk '{print $1}' | sort
else
    perl get_connector_grams.pl 4 $candsUni $corpusTok | \
        grep -v _ | sort | uniq -c | \
        perl -ane 'if ($F[0]>2){print join " ",@F[2...($#F-1)];print
    "\\n";}' | \
        sort | uniq -c | awk '{print $2,$3,$1}' | sort -nrk3 | \
        perl get_top_percentage.pl $percent - | awk '{print $1,$2}' | sort
fi
```

### extract\_bigram.sh

```
#!/bin/sh

#por enquanto, soh ateh trigramas

help () {
    echo "--> Ferramenta:      Gera lista de bigramas"
    echo "      Modo de uso:      $0 <bitok_corpus> <lista_stopwords>
    <arq_candidatos_unitermos>"
    echo "      Dependecies(1):  print_good_ngrams.pl"
    echo "      Resultado:      a lista de candidatos a bigramas coletados no
    corpus"
}

if ! ( [ $1 ] && [ $2 ] && [ $3 ] ); then
    help
    exit
fi
```

```
#clarificando as variaveis
biTok_corpus=$1
stopwordList=$2
candsUniterm=$3

awk '/curr_doc/{print "_stop_"} $0 !~/curr_doc/{print}' $biTok_corpus | \
perl print_good_ngrams.pl 3 $stopwordList $candsUniterm - | sort | \
uniq -c | perl -ane '$fq = shift @F; print join(" ",@F); print " $fq\n";'
```

### **extract\_multi.sh**

```
#!/bin/sh

help () {
    echo "--> Ferramenta:      Extrator de termos multi-palavras"
    echo "      Modo de uso:      $0 <unigramas_conectores> <bigramas_conectores>
<lista_candidatos_ngramas> <top_bigramas> <k>"
    echo "      Dependecies(1):    collect_mw_terms.pl"
    echo "      Resultado:         a lista de termos multi-word extraida do corpus"
}

if ! ( [ $1 ] && [ $2 ] && [ $3 ] && [ $4 ] && [ $5 ] ); then
    help
    exit
fi

#clarificando as variaveis
uni_connectors=$1
bi_connectors=$2
allGrams=$3
top_bigrams=$4
k=$5

awk '(NF>3){print}' $allGrams | perl -ane 'if ($F[$#F]>4){print}' >
XXXTMP_top_morethanbi_grams_XXX

perl -ne 's/ /_/;print' $uni_connectors $bi_connectors| \
perl collect_mw_terms.pl $k - XXXTMP_top_morethanbi_grams_XXX $top_bigrams
rm XXXTMP_top_morethanbi_grams_XXX
```

## Apêndice 4: Lista de Stopwords

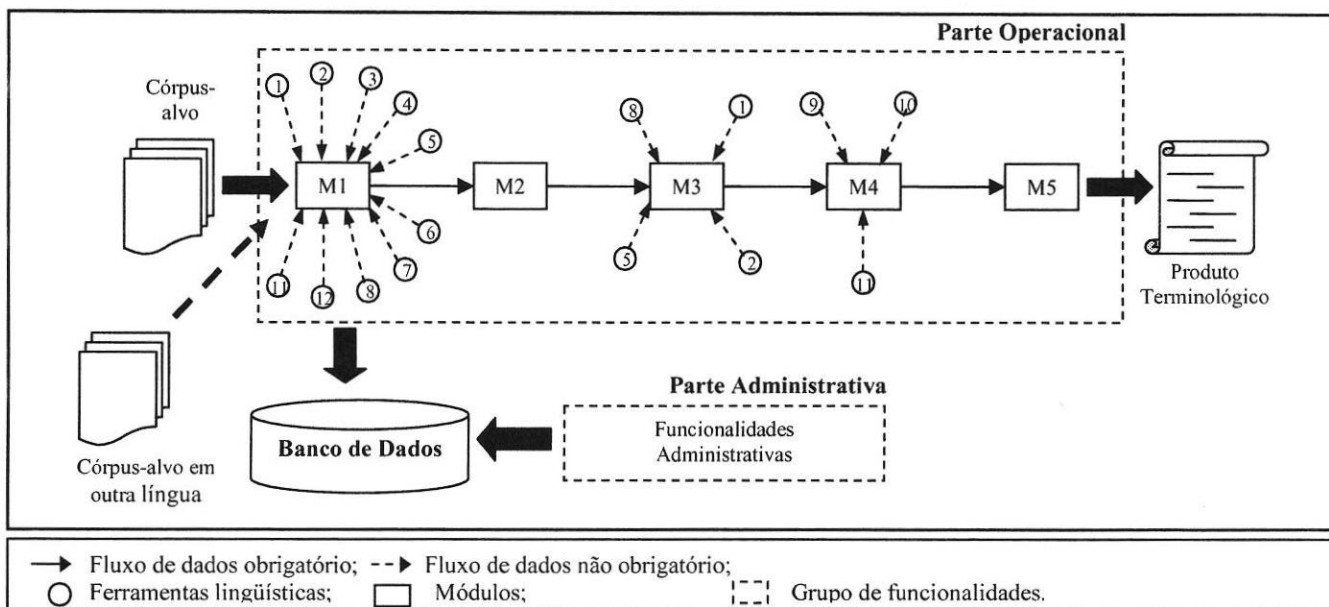
|             |             |            |               |            |
|-------------|-------------|------------|---------------|------------|
| a           | anybody     | beforehand | consequently  | enough     |
| able        | anyhow      | behind     | consider      | entirely   |
| about       | anyone      | being      | considering   | especially |
| above       | anything    | believe    | contain       | et         |
| abstract    | anyway      | below      | containing    | etc        |
| according   | anyways     | beside     | contains      | even       |
| accordingly | anywhere    | besides    | conclusion    | ever       |
| across      | apart       | best       | corresponding | every      |
| actually    | appear      | better     | could         | everybody  |
| acknowledge | appreciate  | between    | course        | everyone   |
| after       | appropriate | beyond     | currently     | everything |
| afterwards  | apendix     | both       | d             | everywhere |
| again       | are         | brief      | definitely    | ex         |
| against     | around      | but        | described     | exactly    |
| all         | as          | by         | despite       | example    |
| allow       | aside       | c          | did           | except     |
| allows      | ask         | came       | different     | f          |
| almost      | asking      | can        | do            | far        |
| alone       | associated  | cannot     | does          | few        |
| along       | at          | cant       | doing         | fifth      |
| already     | available   | cause      | done          | first      |
| also        | away        | causes     | down          | figure     |
| although    | awfully     | certain    | downwards     | fig;       |
| always      | b           | certainly  | during        | fig.       |
| am          | be          | changes    | e             | five       |
| among       | became      | chapter    | each          | followed   |
| amongst     | because     | clearly    | edu           | following  |
| an          | become      | co         | eg            | follows    |
| and         | becomes     | com        | eight         | for        |
| another     | becoming    | come       | either        | former     |
| annex       | been        | comes      | else          | formerly   |
| any         | before      | concerning | elsewhere     | forth      |

|             |              |          |              |           |
|-------------|--------------|----------|--------------|-----------|
| foreword    | hereupon     | j        | meanwhile    | nothing   |
| four        | hers         | just     | merely       | novel     |
| from        | herself      | k        | might        | now       |
| further     | hi           | keep     | more         | nowhere   |
| furthermore | him          | keeps    | moreover     | o         |
| g           | himself      | kept     | most         | obviously |
| get         | his          | know     | mostly       | of        |
| gets        | hither       | knows    | much         | off       |
| getting     | hopefully    | known    | must         | often     |
| given       | how          | l        | my           | oh        |
| gives       | howbeit      | last     | myself       | ok        |
| go          | however      | lately   | n            | okay      |
| goes        | i            | later    | name         | old       |
| going       | ie           | latter   | namely       | on        |
| gone        | if           | latterly | nd           | once      |
| got         | ignored      | least    | near         | one       |
| gotten      | immediate    | less     | nearly       | ones      |
| greetings   | in           | lest     | necessary    | only      |
| h           | inasmuch     | let      | need         | onto      |
| had         | inc          | like     | needs        | or        |
| happens     | indeed       | liked    | neither      | other     |
| hardly      | indicate     | likely   | never        | others    |
| has         | indicated    | little   | nevertheless | otherwise |
| have        | indicates    | look     | new          | ought     |
| having      | inner        | looking  | next         | our       |
| he          | insofar      | looks    | nine         | ours      |
| hello       | instead      | ltd      | no           | ourselves |
| help        | into         | m        | nobody       | out       |
| hence       | introduction | mainly   | non          | outside   |
| her         | inward       | many     | none         | over      |
| here        | is           | may      | noone        | overall   |
| hereafter   | it           | maybe    | nor          | own       |
| hereby      | its          | me       | normally     | p         |
| herein      | itself       | mean     | not          | part      |

|              |           |            |            |               |
|--------------|-----------|------------|------------|---------------|
| particular   | says      | sorry      | therefore  | unfortunately |
| particularly | second    | specified  | therein    | unless        |
| per          | secondly  | specify    | theres     | unlikely      |
| perhaps      | see       | specifying | thereupon  | until         |
| placed       | seeing    | still      | these      | unto          |
| please       | seem      | sub        | they       | up            |
| plus         | seemed    | such       | think      | upon          |
| possible     | seeming   | sup        | third      | us            |
| presumably   | seems     | sure       | this       | use           |
| preface      | seen      | t          | thorough   | used          |
| probably     | self      | table      | thoroughly | useful        |
| provides     | selves    | tab;       | those      | uses          |
| q            | sensible  | tab.       | though     | using         |
| que          | sent      | take       | three      | usually       |
| quite        | serious   | taken      | through    | uucp          |
| qv           | seriously | tell       | throughout | v             |
| r            | seven     | tends      | thru       | value         |
| rather       | several   | th         | thus       | various       |
| rd           | shall     | than       | to         | very          |
| re           | she       | thank      | together   | via           |
| really       | should    | thanks     | too        | viz           |
| reasonably   | since     | thanx      | took       | vs            |
| regarding    | six       | that       | toward     | w             |
| regardless   | so        | thats      | towards    | want          |
| regards      | some      | the        | tried      | wants         |
| relatively   | somebody  | their      | tries      | was           |
| respectively | somehow   | theirs     | truly      | way           |
| right        | someone   | them       | try        | we            |
| s            | something | themselves | trying     | welcome       |
| said         | sometime  | then       | twice      | well          |
| same         | sometimes | thence     | two        | went          |
| saw          | somewhat  | there      | u          | were          |
| say          | somewhere | thereafter | un         | what          |
| saying       | soon      | thereby    | under      | whatever      |

|            |           |         |         |            |
|------------|-----------|---------|---------|------------|
| when       | whereupon | whole   | within  | yet        |
| whence     | wherever  | whom    | without | you        |
| whenever   | whether   | whose   | wonder  | your       |
| where      | which     | why     | would   | yours      |
| whereafter | while     | will    | would   | yourself   |
| whereas    | whither   | willing | x       | yourselves |
| whereby    | who       | wish    | y       | z          |
| wherein    | whoever   | with    | yes     |            |

**Anexo 1:** Ambiente Web colaborativo do Projeto *Extração automática de termos e elaboração colaborativa de terminologias para o intercâmbio de conhecimento especializado*



**Figura 5: Estrutura do Ambiente Web Colaborativo**

Na Figura 5, o Módulo 1 (M1) é responsável pelo suporte e análise da qualidade dos cópus-alvo; o Módulo 2 (M2) pela Extração automática de termos; o Módulo 3 (M3) pela Edição da estrutura conceitual e categorização de termos; o Módulo 4 (M4) pelo Gerenciamento da base de dados terminológicos e o Módulo 5 (M5) pelo Intercâmbio e difusão de termos. A Tabela 10 mostra (de acordo com os círculos numerados na Figura 1) o conjunto de ferramentas de PLN que serão utilizadas no Ambiente do projeto Extração automática de termos e elaboração colaborativa de terminologias para o intercâmbio de conhecimento especializado, fazendo com que ele se diferencie dos demais.

| Número | Nome da Ferramenta                                                                  |
|--------|-------------------------------------------------------------------------------------|
| 1      | Contadores de frequência de palavras                                                |
| 2      | Contadores de frequência de uma única palavra ou expressão                          |
| 3      | Concordanceadores                                                                   |
| 4      | Identificação e separação de lexias complexas                                       |
| 5      | Identificação e recuperação de termos do corpus                                     |
| 6      | Etiquetadores (taggers)                                                             |
| 7      | Segmentador Sentencial                                                              |
| 8      | Lematizadores                                                                       |
| 9      | Corretores gramaticais (identificação de erros ortográficos e sintáticos – parsers) |
| 10     | Editores de definição (tipologia de definição)                                      |
| 11     | Alinhador de palavras                                                               |
| 12     | Identificação de Unigrama, Bigramas e Trigramas                                     |

**Tabela 10: Conjunto de ferramentas de PLN**

## Anexo 2: Interface Web do Ambiente e-Termos apresentado o Módulo de criação de corpus descartáveis implementado com o BootCat

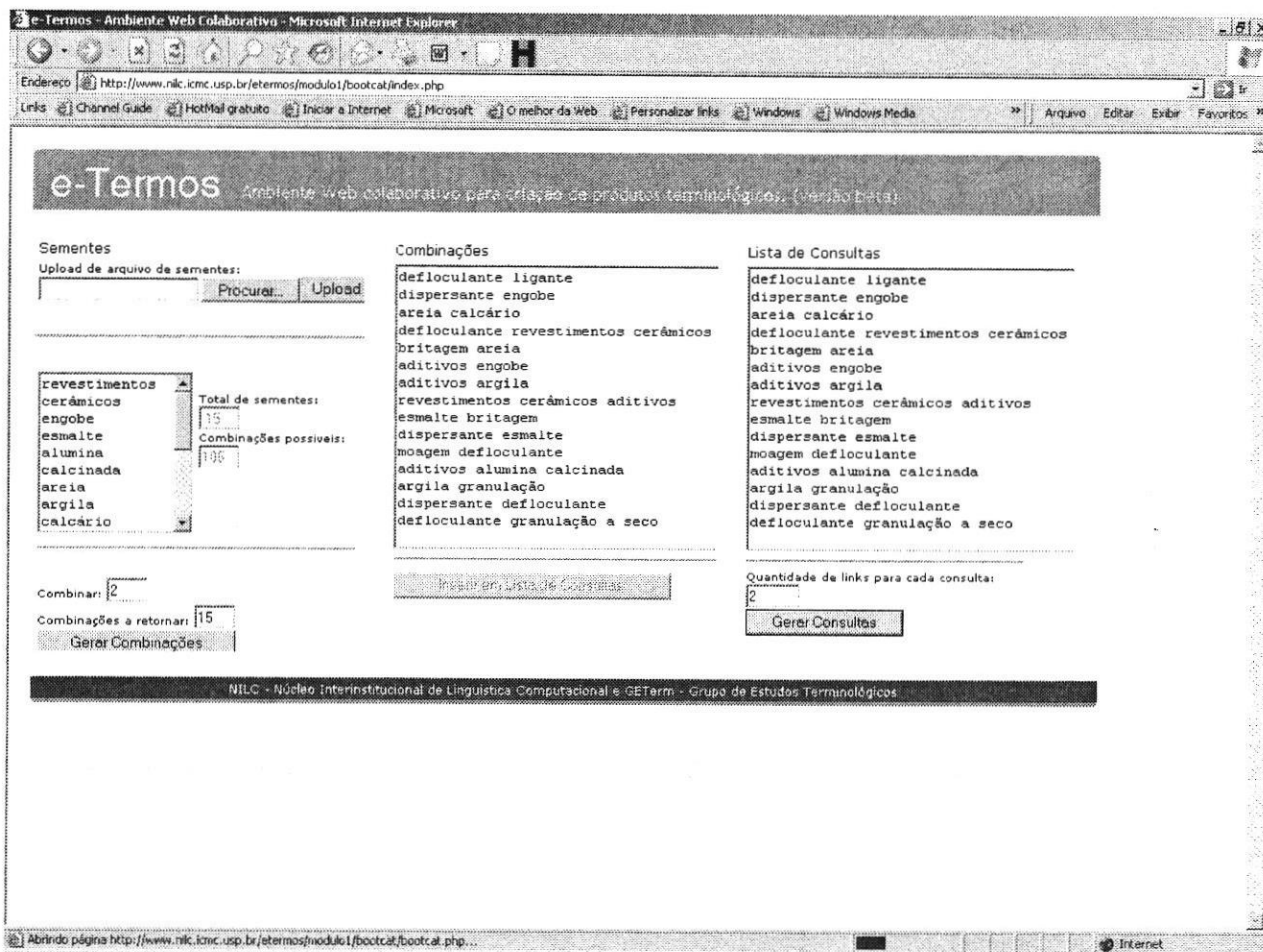
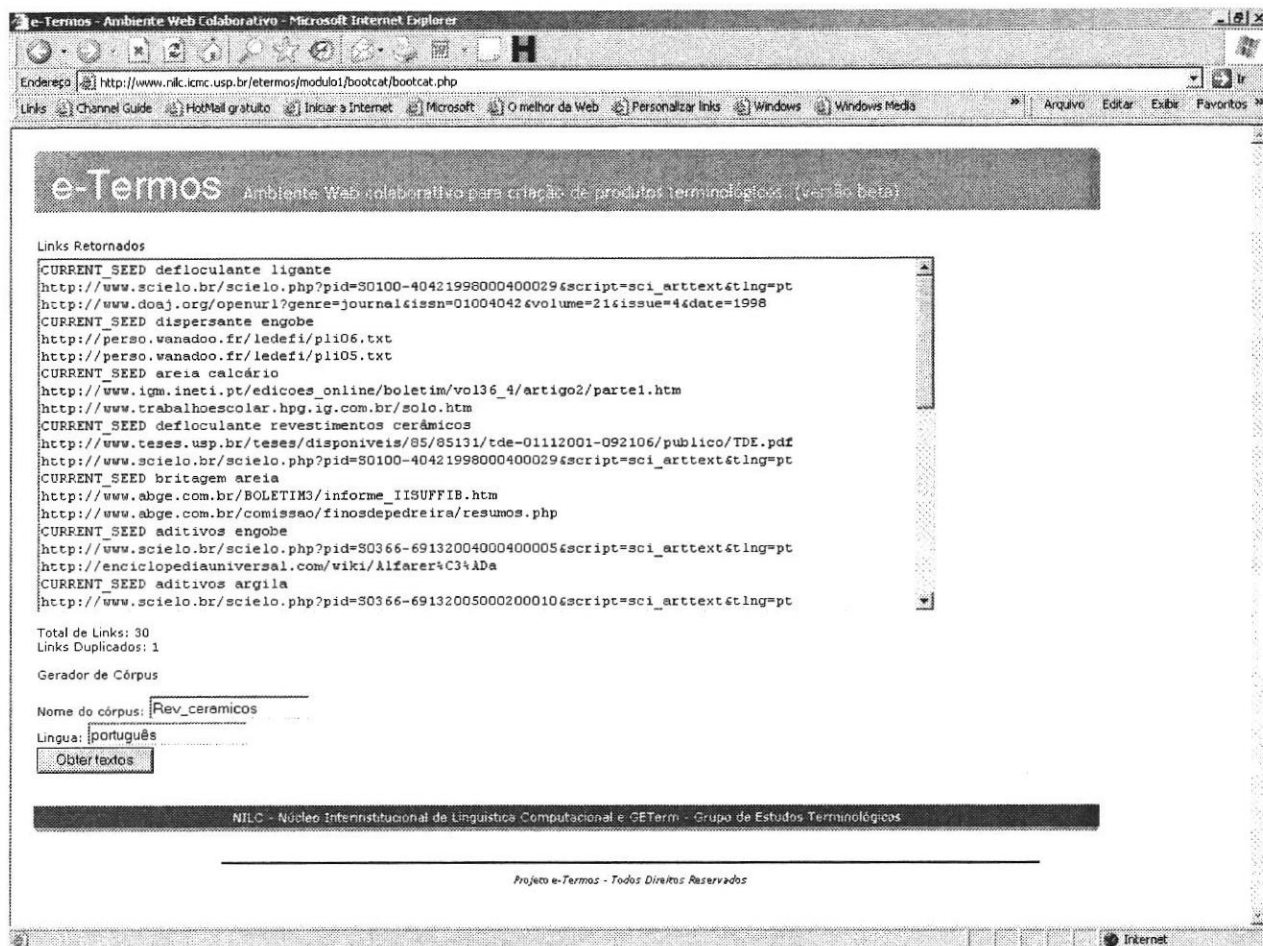
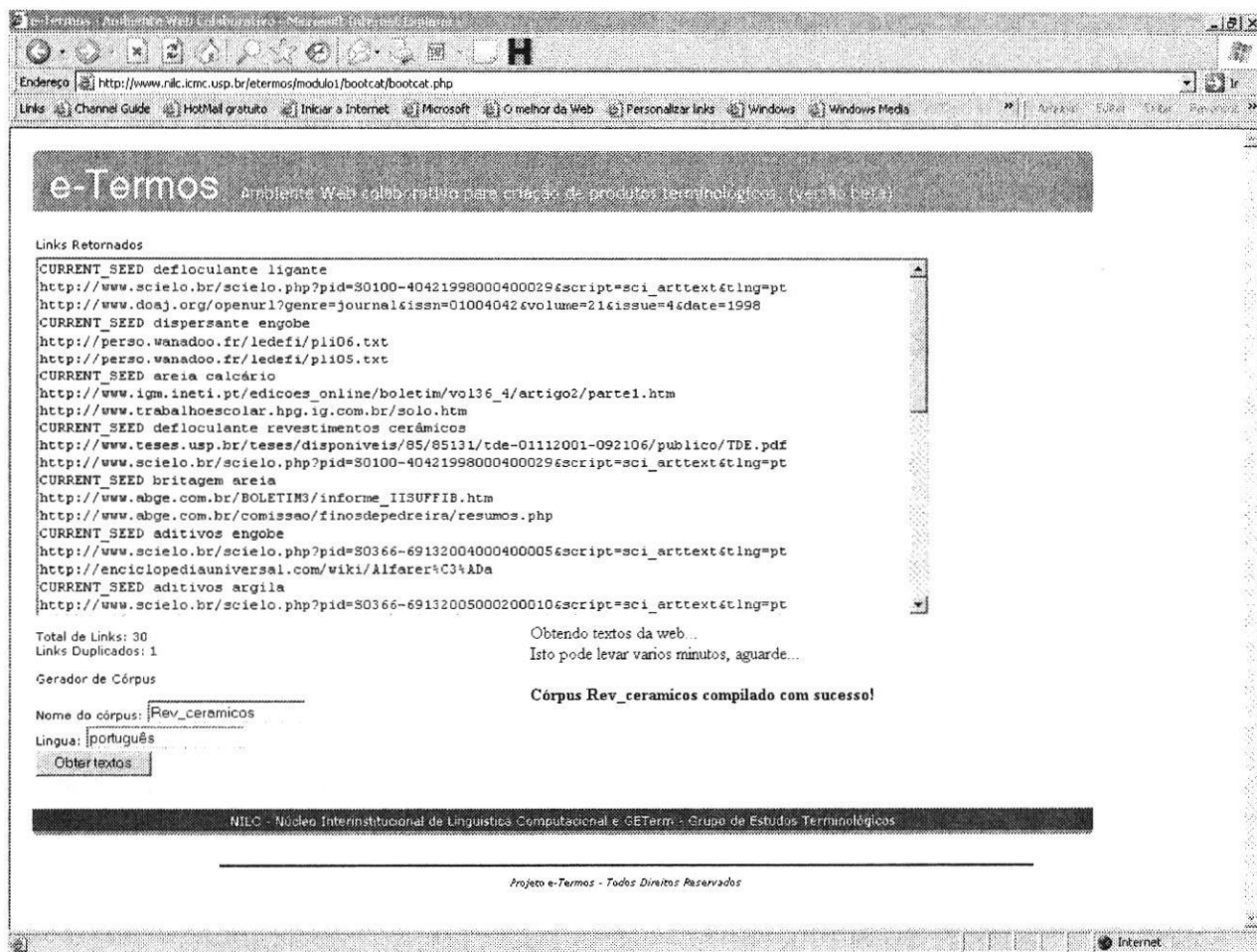


Figura 6: Tela inicial do Processo de Geração de um Corpus Descartável para Pesquisas Terminológicas. Essa tela permite fazer o **upload** de um arquivo com sementes (os autores do BootCat recomendam de 5 a 15 sementes que são termos discriminantes do domínio de pesquisa); no exemplo da figura foram escolhidas 15 sementes e pedida a combinação 2 a 2 com elas -- o que daria 105 combinações no máximo -- e para retornar 15 combinações desse conjunto de 105. O domínio escolhido foi o de Revestimentos Cerâmicos. Na segunda janela ("Combinações") são apresentadas as 15 combinações com 2 termos escolhidas e na terceira janela ("Lista de Consultas") as combinações que serão passadas como buscas ao Google. Escolhe-se também a quantidade de links a serem gerados para cada consulta. No caso da figura foram escolhidos 2 links.



**Figura 7. Segunda tela do Processo de Geração de um Corpus Descartável para Pesquisas Terminológicas.** Essa tela mostra o total de links retornados e o número de duplicados. Pede o nome do corpus a ser criado e a língua. Ao clicar no botão “Obter Textos” o processo de obtenção de textos da Web com o Google se inicia.



**Figura 7: Corpus compilado com sucesso. O corpus Rev\_ceramicos fica então disponível para os próximos módulos do ambiente e-Termos mostrado esquematicamente no Anexo 1.**

## Glossário

**Token:** uma palavra, um segmento de texto considerado uma palavra.

**Unigrama:** termo composto por 1 token.

**Bigrama:** termo composto por 2 tokens.

**Trigrama:** termo composto por 3 tokens.

**N-grama:** termo composto por N tokens.

**Termo multi-palavra:** termo que envolve 2 ou mais tokens.

**Script:** Um conjunto de comandos escrito em uma linguagem interpretada, como Perl ou Bash.