Projecto TerminUM

J. J. Almeida **Alberto Simões** **José Castro** **Bruno Martins** **Paulo Silva**

Resumo *O projecto TerminUM tem como objectivos principais o estudo, experimentação e a criação de recursos na área dos corpora paralelos, terminologia (descritiva) e recursos multilingues ligados a corpora.*

- *fazer extracção tão automática quanto possível de corpora a partir da web;*
- *fazer extracção de dicionários, de terminologia e de outros recursos ligados à tradução;*
- *criar e interligar as ferramentas desenvolvidas;*
- *criar e disponibilizar: (1) listas de Bitextos, corpora e corpora paralelos, (2) ferramentas de criação e transformação de corpora, (3) recursos multilingues derivados/ligados a corpora.*

Nesta apresentação serão abordadas algumas tarefas presentemente a decorrer no âmbito do projecto, nomeadamente:

1. *ciclo de vida da construção e transformação de corpora;*
2. *resumo das ferramentas desenvolvidas (e em desenvolvimento);*
3. *construção de corpora paralelos tomando como base legendas de filmes (subtitles), ficheiro de internacionalização (mensagens de software .po) e ficheiros de memórias de tradução (TMX);*
4. *animação de corpora paralelos via web (criação de motores de consulta usando diversas ferramentas).*

1 Introdução

É universalmente reconhecido que os recursos multilingue são cruciais para várias áreas ligadas a estudos da língua, como sejam a área da tradução, a extracção de terminologia ou a criação de dicionários. No entanto estes recursos são normalmente difíceis de obter e organizar.

O projecto TerminUM tem como objectivo a recolha, tratamento e disponibilização de recursos bilingues na (e da) Internet. Para isso foi definido um grafo de processos que descreve como obter recursos paralelos a partir de documentos obtidos de uma variedade de origens incluindo-se aqui também documentos extraídos da Internet. A figura 1 mostra os vários processos intervenientes no TerminUM.

As secções seguintes resumem alguns dos processos apresentados no diagrama. Este artigo irá centrar-se especialmente no processo de "preparação de corpora" e "alinhamento à frase", apresentados na secção 2, e nos processos de criação de corpora de legendas de mensagens de programas (.po).

1.1 Recolha Web

Para ser possível a recolha de textos paralelos na Internet torna-se necessário encontrá-lo. Existem vários métodos para realizar esta operação. Um dos métodos, apresentado em [8, 7], utiliza um motor de pesquisa na Internet para encontrar páginas em línguas diferentes que se relacionem de forma particular.

Um outro método, bastante mais leve consiste em analisar um conjunto de endereços de Internet (URL's) e usar heurísticas sobre os nomes dos ficheiros para descobrir relações entre eles. De facto, é habitual que na construção de *sites* em várias línguas se atribuam nomes elucidativos às pastas ou ficheiros em causa.

Finalmente, podemos obter textos paralelos de um *site* que conhecemos e que sabemos conter textos paralelos. Nesse caso, podemos ir buscar todo o *site* e utilizar métodos de comparação entre os ficheiros para encontrar de forma automática a relação entre ficheiros.

No final deste processo obtém-se pares de ficheiros candidatos a bitextos que vamos validar e confirmar se são traduções das línguas que andamos à procura.

1.2 Validação de candidatos a bitextos

Aos pares candidatos é aplicado um processo de validação. Os métodos automáticos de recolha não nos garantem um conjunto de propriedades destes textos. O processo de validação verifica:

- se as línguas dos ficheiros em causa são as pretendidas;
- se os ficheiros têm ambos o mesmo tipo, e se o tipo é reconhecido;
- se os ficheiros têm tamanhos compatíveis;

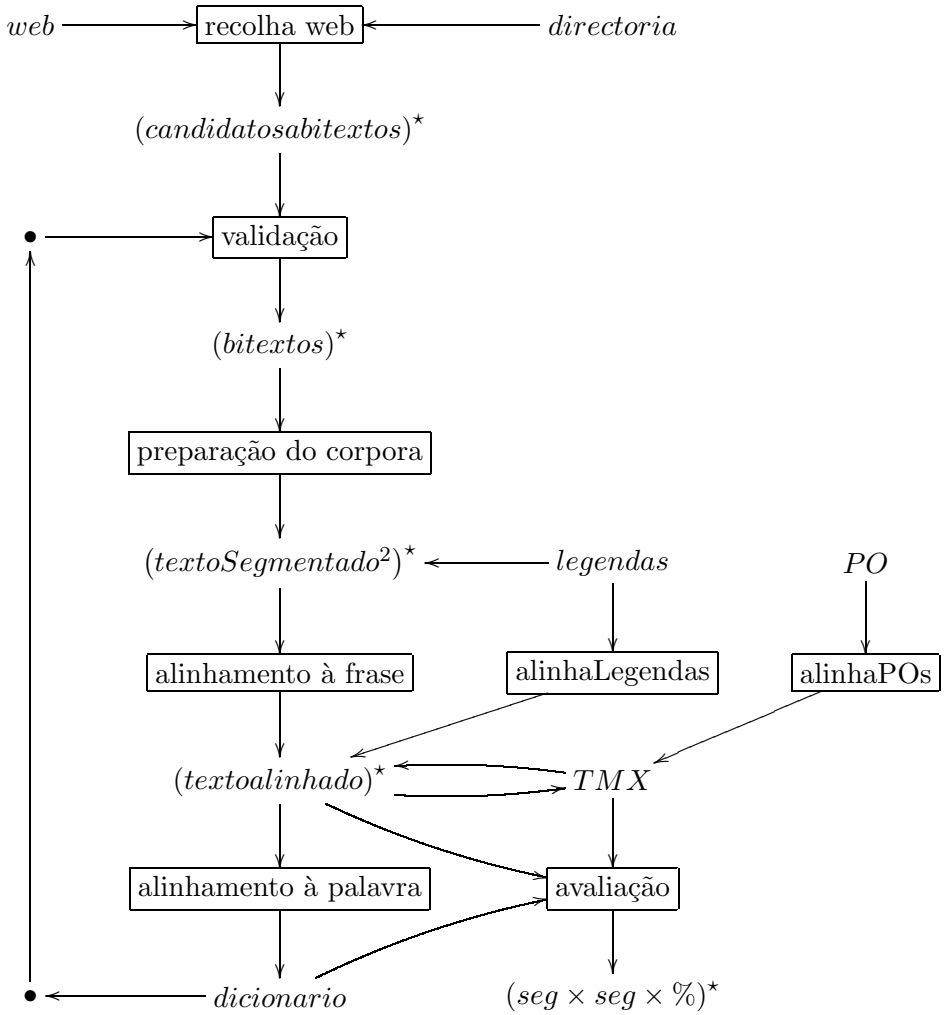

Figura 1: Ciclo de vida e diagrama de construção TerminUM

- se os nomes dos ficheiros são semelhantes;
- se o conteúdo não textual é parecido entre os dois ficheiros (como a pontuação, números, comandos);

Aos pares que passarem este processo de validação iremos chamar **bitextos**.

1.3 Preparação dos corpora

Dos ficheiros recolhidos é necessário remover a marcação específica do formato de ficheiro, limpando tudo o que não seja interessante para a criação do corpus. Este processo é dependente do formato em que o ficheiro se encontra.

Depois de limpo, o texto é segmentado e etiquetado num formato XML que denominamos por PML, em que as etiquetas só assinalam os segmentos obtidos.

Posteriormente, cada ficheiro PML é inserido no *IMS Corpus Workbench* (CWB)[6] para permitir maior eficiência em consultas subsequentes. Este processo pode ser complementado com a lematização dos corpora utilizando um analisador morfológico.

1.4 Alinhamento à frase

Depois de criados, estes corpora podem ser alinhados à frase. Para isso é utilizado o *easyalign*, ferramenta que faz parte do CWB. Estes corpora alinhados podem ser consultados como corpora independentes ou de forma síncrona.

Existe, também, a possibilidade de converter corpora paralelos em ficheiros Translation Memory Exchange (TMX) e vice-versa, para permitir o seu uso por terceiros, como seja a comunidade da tradução.

1.5 Alinhamento à palavra

Depois de alinhados à frase, os corpora podem ser alinhados à palavra. Para isso, estamos a utilizar um conjunto de ferramentas denominado NATools (Natura Alignment Tools). Estas ferramentas são

baseadas no alinhador desenvolvido por Hiemstra [5, 4] no âmbito do projecto Agenda 21 da Comunidade Europeia.

Este alinhamento permite obter dicionários entre as duas línguas em questão, em que a cada palavra de uma das línguas é associado um conjunto de possíveis traduções e suas respectivas probabilidades de tradução.

1.6 Validação de bifrases

Estes mesmos dicionários podem ser usados posteriormente para enriquecer o processo de validação de traduções. De facto, uma medida de avaliação de traduções pode ser obtida calculando, para cada palavra de uma frase se uma das suas possíveis traduções se encontra na outra frase. A média pesada destes valores dá um estimador da correcção da tradução.

Este estimador está a ser usado para filtrar memórias de tradução de forma a obter apenas as unidades que têm qualidade superior a determinado valor, e para ajudar a afinar os processos descritos em 1.2 e 1.4.

2 Criação e disponibilização de corpora

Nesta secção descreve-se algumas das sub-tarefas ligadas à construção de corpora bilingues consultáveis via *web* e na construção das suas versões em formato TMX.

Mais uma vez, cada processo de transformação pode ser feito de diversos modos, sendo no entanto fornecido pelo projecto TerminUM um comando que o execute sem qualquer acção interactiva, de modo a permitir composição dos vários comandos num único, e de modo a permitir o tratamento de grandes quantidades de informação. Em vários dos processos recorreu-se às ferramentas do CWB normalmente integradas com alguns programas que fazem as necessárias adaptações de formatos.

Na sequência do anteriormente descrito, vamos descrever um conjunto de tarefas que:

- fazem a limpeza e normalização dos ficheiros;
- constroem informação adicional: juntar a cada palavra os possíveis lemas e `POS`;
- alinhamento à frase;
- tradução de e para memórias de tradução (TMX);
- disponibilização para consulta via *cgi-web*[3, 10, 9];

2.1 Conversão de ficheiros HTML em PML

No processo de normalização e limpeza dos ficheiros, há que retirar certa informação contida nos textos HTML, dividir os mesmos em frases (segmentação) e convertê-los para um formato comum, usado nos passos seguinte: PML.

O formato PML não é mais que simples texto com marcas XML para separar diferentes ficheiros — `<f id="num" name="nome">...</f>` — e para separar períodos — `<p>...</p>`. Embora estejamos a usar uma etiqueta existente em HTML, não incluímos só parágrafos mas um conjunto de unidades mais vasto como sejam títulos, legendas, partes de tabelas e outros sub-elementos por vezes designados de unidades de tradução.

O atributo "`id`" do elemento "`f`" é usado para sincronização no processo de alinhamento à frase.

A separação em unidades de tradução está a ser guiada por uma tabela que divide as etiquetas HTML em:

- etiquetas a remover, preservando o seu conteúdo (Ex. as etiquetas `html, em, i, u, body`);
- etiquetas a remover bem como o respectivo conteúdo (Ex. as etiquetas `frameset, head, meta, script`);
- etiquetas separadoras de unidade de tradução (Ex. as etiquetas `h1, li, p, blockquote`);
- etiquetas a conservar.

O conjunto de etiquetas em cada classe pode ser alterado usando uma série de opções disponíveis.

Após feita a divisão em unidades de tradução de acordo com a tabela anterior, é ainda feita a segmentação de cada unidade, usando um segmentador tradicional (incluído no módulo Lingua::PT::pln).

2.2 Lematização de ficheiros PML e sua indexação com CWB

Durante esta fase do processo, é feito o cálculo dos possíveis lemas e *part-of-speech* (POS) de cada palavra e conversão para o formato CWB.

O cálculo de lemas e POS está a usar a biblioteca perl do Jspell [1, 11] e produz lemas e POS ambíguos: cada palavra pode dar origem a vários lemas e vários POS. As várias hipóteses encontradas estão a ser formatadas de acordo com o formato esperado pelos atributos ambíguos em CWB.

Seguidamente é determinado qual o conjunto de etiquetas usadas e é construído um corpus CWB, usando os seus indexadores com um conjunto de opções determinadas automaticamente.

2.3 Alinhamento à frase

O alinhamento à frase é feito usando o *easyalign* — um alinhador à frase que faz parte do CWB. Este alinhador funciona sem qualquer tipo de interactividade — o que é crucial neste processo, devido aos tamanhos envolvidos.

Havendo (pelo menos) duas línguas em análise são feitos alinhamentos em ambos os sentidos.

2.4 Conversão de e para memórias de tradução (TMX)

A conversão para TMX foi ditada pela vontade de realizar intercâmbios com outras comunidades como seja a comunidade de tradução. Deste modo permitimos que:

- outras pessoas possam produzir recursos a serem utilizados no TerminUM;
- certas ferramentas ligadas à tradução possam ser usadas sobre os corpora criados no projecto;
- um conjunto mais vasto de pessoas possa validar implicitamente o trabalho realizado.

A conversão para TMX está a ser realizada com uma mistura de navegação no corpus paralelo CWB com uma geração do texto XML ligado à TMX (`XML::TMX`).

A conversão TMX para CWB está a usar um reconhecedor de XML e a fazer uma transformação estrutural para PML com atributos de alinhamento, sendo depois reutilizados os conversores descritos anteriormente.

2.5 Criação de corpora consultáveis via *web*

Uma das questões a que se atribuiu grande importância foi à construção de mecanismos de consulta remota. Deste modo, estamos a construir recursos que são úteis à comunidade e ao mesmo tempo, estamos a conseguir que os recursos bilingues sejam implicitamente testados.

Os corpora ficam consultáveis via *web* à custa de um programa (CGI) que dinamicamente:

- determina quais os corpora disponíveis no sistema;
- determina quais deles são paralelos;
- constrói uma página HTML que permite escolher o corpus a usar (bem como qual o tamanho do contexto a apresentar) aceitar expressões de pesquisa CQP-CWB e mostrar as respostas (ver figura 2).

2.6 Sistema geral em funcionamento

Se dispusermos de um ficheiro `F.pairs` contendo em cada linha um par de nomes de ficheiros que sejam a tradução um do outro:

```
f1_pt.html    f1_eng.html
f2_pt.html    f2_eng.html
...
```

e executarmos o comando

```
mkterminum F.pairs
```

todos os recursos referidos nesta secção ficam calculados, incluindo o facto de que o corpus fica consultável na Internet.

Para além deste comando geral, todas as etapas anteriores estão acessíveis com comandos individuais, permitindo realizar as respectivas tarefas de modo mais controlado.

3 Corpus de legendas (subtitiles)

Nesta secção apresenta-se a experiência ligada à criação de um corpus paralelo Subterminum de legendas de filmes e à criação das ferramentas necessárias ao seu enquadramento no grafo de processos da figura 1.

3.1 Introdução

Com o advento dos filmes em DVD, cada filme vem legendado em várias línguas. Da mesma forma surgiram formatos para as codificar bem como *sites* na Internet que os disponibilizam gratuitamente.

3.1.1 Formatos de subtitiles

Há uma grande variedade de formatos usados em legendas, sendo alguns deles específicos de aplicações capazes de mostrar vídeos. Os mais habituais são os formatos SubRip (extensão `.srt`), MicroDVD (extensão `.sub`) e SubViewer.

Um ficheiro de legendas, simplificadamente, é constituído por uma série de frases, associadas a um instante de aparecimento e outro de desaparecimento. Esses instantes podem ser descritos por uma medida de tempo em segundos, como é o caso do formato SubRip (**srt**):

```
36
00:02:56,252 --> 00:02:59,481
Vai ver muito mais coisas deste tipo no filme.

37
00:02:59,549 --> 00:03:01,272
Coisas que aconteceram de verdade.

...
```

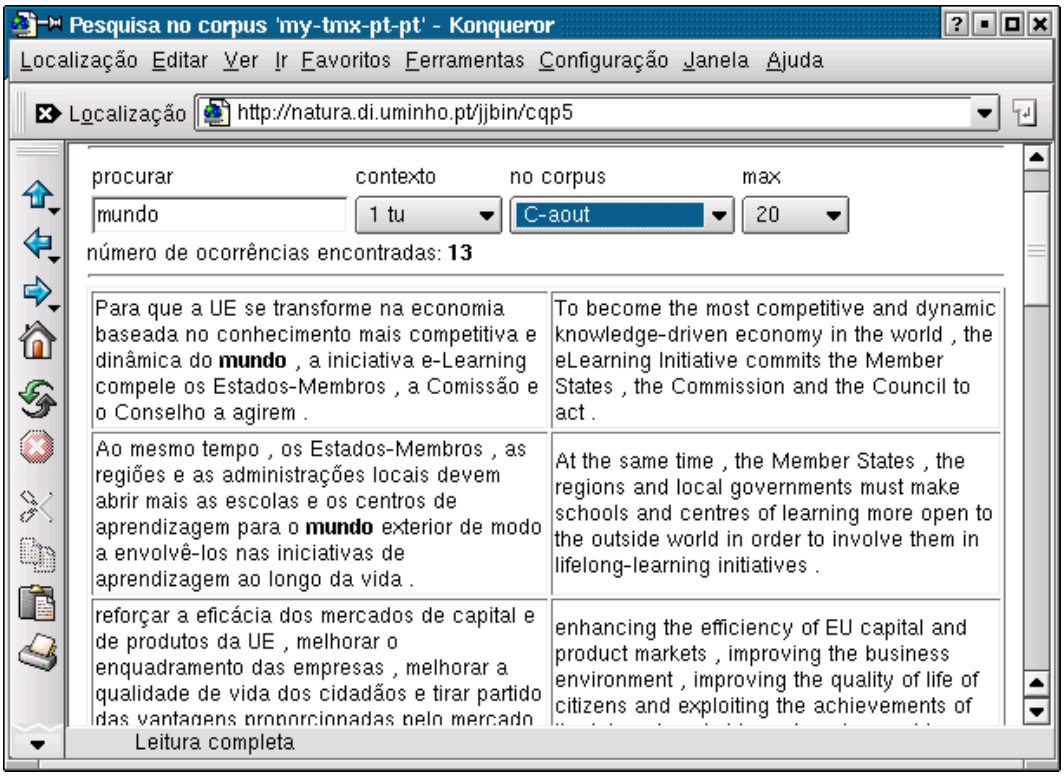

Figura 2: Pesquisa em corpora paralelos usando a CGI de pesquisa

ou por um medida em termos de número de *frames* a partir do início do filme, como é o caso do formato de legendas MicroDVD (**sub**):

```
{40643}{40714}So then, who pays?
{40718}{40776}The Board of Education.
{40780}{40830}- Lord, did you hear that?
{40869}{40946}- Sound good?|- Hey!
...
```

O número de frames está implicitamente ligado à velocidade com que as frames são passadas, velocidade essa que pode ser diferente conforme o ficheiro, sendo o valor habitual de 25 *frames* por segundo.

3.1.2 Problemas habituais com subtitiles

As legendas são frequentemente constituídas por amadores podendo ter qualidade muito variada.

Às vezes surgem problemas de os ficheiros de legendas estarem com a extensão errada, ou de algum modo estarem com partes corrompidas.

Do ponto de vista do seu conteúdo é normal haver partes que só aparecem numa das línguas (exemplo: a tradução de um letreiro que aparece na cena, ou a inclusão de um poema de uma canção que esteja a ser cantada).

3.1.3 Descrição do corpus Subterminum

Juntaram-se legendas (basicamente referentes às línguas portuguesa, inglesa, castelhana, portuguesa do Brasil, e francesa) de cerca de 100 filmes.

Esse conjunto de legendas foi armazenado sistematicamente numa árvore de directorias (uma directoria por filme; dentro de cada directoria está guardado o conjunto de ficheiros de legendas tendo como nome o identificador da língua correspondente) e um ficheiro com meta-informação acerca do filme[1] e colocado em CVS.

3.2 Alinhamento meramente textual de subtitles

Este tipo de alinhamento baseia-se em converter as mensagens para PML (em que cada unidade de tradução `<p>` marca uma legenda), e usar o alinhamento normal sobre bitextos (usando a ferramenta `xmlalign2cqp` que converte os bitextos para CWB, faz o alinhamento e cria também um corpus paralelo pesquisável pela WEB).

Para isso foi construído um conversor `subtitle2pml` que para um par de línguas faz a travessia da árvore de filmes e para cada filme que tenha legendas nas duas línguas pretendidas, detecta o formato de legenda usado (de momento só reconhece `.srt` e `.sub`) e gera PML. O nome do filme serve de elemento de sincronização.

Na criação dum corpus Subterminum pt-en foi

[1] Descrevendo o tipo de filme (Ex: desenhos animados infantis, coboiada, ficção científica, etc), origem, data, etc.

necessário invocar os seguintes comandos[2]

comando	tempo
subtitle2pml -l1=pt -l2=en	0m15.94s
xmlalign2cqp filmes_pt filmes_en	16m13.47s

O corpus criado nesta experiência tem o seguinte tamanho:

	Português	Inglês
unidades de tradução	126544	140136
palavras	526298	611544
tamanho em bytes	3602608	3930725

O facto de todo o processo ser automático, torna viável que seja feita uma variedade de corpora paralelos para os diferentes pares de línguas, bem como permite que se continue a fazer crescer o conjunto de filmes.

3.3 Alinhamento temporal-textual de subtitiles

Este modo de alinhamento leva em conta a informação temporal contida nos ficheiro de legendas, permitindo que seja feita uma segmentação em blocos e que se possa ter em conta a taxa de sobreposição temporal para decidir se certas legendas são correspondentes (1:1, 1:2, 1:0).

Para realizar este alinhamento é feito o seguinte conjunto de passos:

1. sincronização de ficheiros (de modo a garantir o mesmo referencial temporal);
2. segmentação;
3. eliminação de segmentos desequilibrados;
4. alinhamento dos segmentos.

3.3.1 Sincronização de subtitiles

Os vários ficheiro de legendas diferem frequentemente na origem do eixo dos tempos (podem incluir inícios mais ou menos extensos). Para que se possa fazer a comparação temporal, torna-se necessário fazer translações (e por vezes *scaling*) no domínio tos tempos de modo a garantir o mesmo referencial.

Normalmente nas traduções de legendas, os nomes dos personagens são preservados. Desta forma essa sincronização está a ser feita procurando os nomes próprios contidos nos ficheiros que tenham igual número de ocorrências e calculando as transformações necessárias de modo a que as legendas que os contenham se sobreponham o melhor possível.

3.3.2 Segmentação

A segmentação das legendas em blocos está a basear-se em espaços de silêncio e em nomes próprios co-ocorrentes.

[2]Medidas tomadas num pentium 600Mhz, 256Mbytes de RAM, Linux em carga

Após ter segmentado as legendas por este processo, torna-se possível a rejeição de blocos demasiado desequilibrados (Exemplo: um bloco que tenha 10 legendas numa das línguas e 1 legenda na outra), eliminando tipicamente extractos de legendas só presentes numa das línguas.

4 Corpus de PO

Os ficheiros PO estão associados ao problema da **internacionalização** dos programas (i.e. fazer com que um programa possa funcionar em várias línguas).

O ficheiro PO de uma determinada língua de um determinado programa, contém identificadores de cada mensagem que aparece nesse programa associados à sua tradução nessa língua.

4.1 Formato PO

Considere-se o seguinte exemplo de um extracto simples de um ficheiro PO referente à lingua portuguesa:

```
msgid  "/users <chan>: list the users on
                                  channel <chan>"
msgstr "/users <canal>: lista os utilizadores
                               no canal <canal>"

msgid  "/version: get the server version"
msgstr "/version: obtém a versão do servidor"

msgid  "/whois: get info about this user"
msgstr "/whois: obtém informações sobre
                                este utilizador"

msgid  "/whois <user>: get info about a user"
msgstr "/whois <user>: obtém informações
                            sobre um utilizador"

msgid  "0 of 0 differences "
msgstr "0 de 0 diferenças "

msgid  "0 of 0 files "
msgstr "0 de 0 ficheiros "

msgid  "0% No Change"
msgstr "0% Sem Alteração"
```

Cada linha com "msgid" identifica a mensagem (neste caso a identificação está a ser feita através da mensagem em Inglês, o que é uma prática comum não obrigatória). Cada linha com "msgstr" define a tradução da mensagem na língua em causa (neste caso Português). Por vezes esta tradução pode estar vazia (ainda não traduzida, fazendo com que o programa mostre a mensagem na língua original).

O extracto correspondente em lingua Castelhana é:

```
msgid  "/users <chan>: list the users on
                                channel <chan>"
msgstr "/users <can>: obtener la lista de
                     usuarios del canal <can>"

msgid  "/version: get the server version"
msgstr "/versión: obtener la versión del
                                    servidor"

msgid  "/whois: get info about this user"
msgstr "/whois: obtener información sobre
                                  este usuario"

msgid  "/whois <user>: get info about a user"
msgstr "/whois <usuario>: obtener información
                                 sobre usuario"

msgid  "0 of 0 differences "
msgstr "0 de 0 diferencias "

msgid  "0 of 0 files "
msgstr "0 de 0 archivos "

msgid  "0% No Change"
msgstr "0% sin cambio"
```

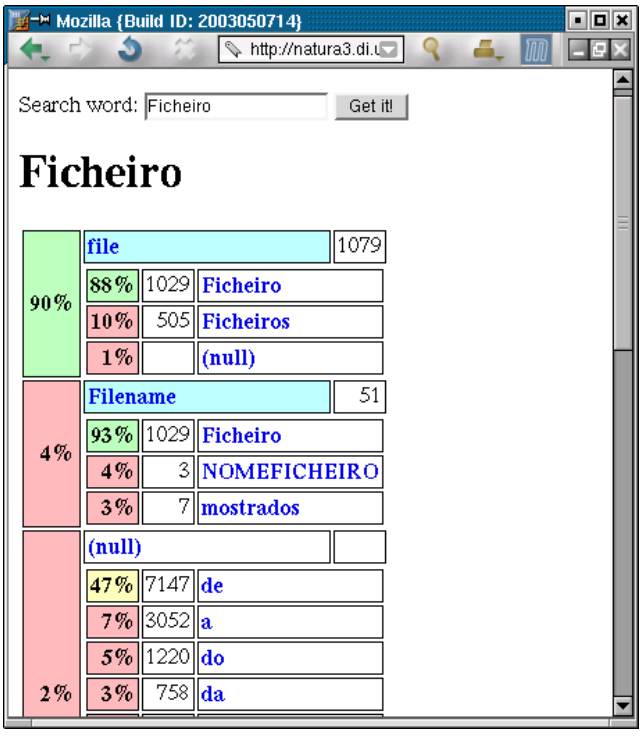

Figura 3: Pesquisa de "ficheiro" no dicionário gerado pelas NATools, a partir dum corpus PO

4.2 Alinhamento de POs

A partir de ficheiros deste tipo, construiu-se um transformador **po2tmx.pl** que analisa os PO de um determinado programa (projecto) e constrói TMXs para os vários pares de línguas que se pretenda.

Para tal, é calculada a correspondência

$$mesid \mapsto (lingua \mapsto megstr)$$

e a partir desta são gerados os ficheiros TMX para as línguas pretendidas (escolhendo somente as entradas em que ambas as línguas estejam definidas)

Com base nas TMXs é fácil a construção de corpora paralelos e seus derivados, usando as ferramentas do projecto, neste caso:

```
1 po2tmx.pl /opt/gnome-2.2/
2 tmx2cqp  -l1=pt -l2=en *_pt_es.tmx
```

onde:

- na linha 1 se faz a travessia de toda a árvore do projecto gnome criando 835 TMXs
- e na linha 2 se converte para formato CWB, criando um corpus paralelo pesquisável via WEB com as 167 TMX referentes a Português-Castelhano.

O mesmo corpus pode ser usado para construir um dicionário NATools ou para qualquer uso.

Na figura 3 aparece a pesquisa da palavra "Ficheiro" num dicionário NATools criado com as duas maiores TMXs geradas a partir dos POs do projecto Gnome.

5 Conclusões e trabalho futuro

Os métodos descritos neste documento permitem obter de forma eficiente corpora bilingue a partir de várias origens. O seu tratamento, embora de forma automática, está a produzir corpora paralelos de qualidade razoável e a disponibilizá-los para pesquisa.

O alinhamento temporal/textual de legendas está ainda em fase experimental mas tem já mostrado ser capaz de produzir resultados úteis.

O facto de se dispor de um um grafo de processos de tradução de formatos e de construção de novos recursos faz com que um problema novo possa ser resolvido mais rapidamente.

As ferramentas referidas neste documento estão disponíveis a partir das páginas do projecto TerminUM `http://natura.di.uminho.pt/terminum`.

Referências

[1] J.J. Almeida and Ulisses Pinto. Jspell – um módulo para análise léxica genérica de linguagem natural. In *Actas do Congresso da Associação Portuguesa de Linguística, Évora*, 1994.

[2] José João Almeida, Alberto Manuel Simões, and José Alves Castro. Grabbing parallel corpora from the web. Number 29, pages 13–20. Sociedade Española para el Procesamiento del Lenguaje Natural, Sep. 2002.

[3] Ana Frankenberg-Garcia and Diana Santos. Apresentando o compara, um corpus português-inglês na web. In *Cadernos de Tradução*.

[4] Djoerd Hiemstra. Multilingual domain modeling in twenty-one: automatic creation of a bi-directional lexicon from a parallel corpus. Technical report, University of Twente, Parlevink Group, 1998.

[5] Djoerd Hiemstra. Using statistical methods to create a bilingual dictionary. Master's thesis, Department of Computer Science, University of Twente, August 1996.

[6] Oliver Christ & Bruno M. Schulze & Anja Hofmann & Esther König. *The IMS Corpus Workbench: Corpus Query Processor (CQP): User's Manual*. Institute for Natural Language Processing, University of Stutgart, March 8, 1999 (CQP V2.2).

[7] Philip Resnik. Parallel strands: A preliminary investigation into mining the web for bilingual text. In *D. Farwell, L. Gerber, and E. Hovy (eds.), Machine Translation and the Information Soup (AMTA-98)*, 1998. Lecture Notes in Artificial Intelligence 1529, Springer.

[8] Philip Resnik. Mining the web for bilingual text. In *37th Annual Meeting of the ACL'99*, 1999. College Park, Maryland.

[9] Paulo Alexandre Rocha and Diana Santos. CETEMPúblico: Um corpus de grandes dimensões de linguagem jornalística portuguesa. In *Actas do V Encontro para o processamento computacional da língua portuguesa escrita e falada (PROPOR'2000)*, pages 131–140, November 2000.

[10] Diana Santos and Eckhard Bick. Providing internet access to portuguese corpora: the AC/DC project. In Maria Gavrilidou et al, editor, *Second International Conference on Language Resources and Evaluation, LREC 2000*, pages 205–210, June 2000.

[11] Alberto Manuel Simões and José João Almeida. `jspell.pm` — um módulo de análise morfológica para uso em processamento de linguagem natural. In *Actas da Associaçáo Portuguêsa de Linguística*, 2001.