Volume 4, Número 2- Dezembro 2012

linguaMATICA

ISSN: 1647-0818

UNIVERSIDADE DE VIGO

Universidade do Minho

Volume 4, Número 2 – Dezembro 2012

LinguaMÁTICA

ISSN: 1647–0818

Conteúdo

Editorial

Esta edição termina o quarto ano de vida da revista Linguamática. O ano foi composto por uma edição especial, relativa a uma avaliação conjunta, o Págico, e esta edição convencional. Sentimo-nos bastante satisfeitos por conseguir nestes tempos manter a revista viva, mesmo que com relativa dificuldade.

A verdade é que cada vez mais investigadores e docentes do ensino superior são avaliados não pela qualidade da sua investigação ou dos resultados obtidos, mas por uma classificação das conferências e revistas.

Embora a Linguamática já se encontre indexada em vários índices, como o Latindex, não se encontra nos índices habitualmente usados, como o DBLP, ISI Web of Knowledge ou SCOPUS.

Será muito complicado conseguir que a Linguamática seja considerada para qualquer um destes índices, especialmente pela língua usada não ser o inglês. No entanto, mudar a política da Linguamática para a publicação em inglês é impensável, já que iria contra os princípios que levaram à sua criação. No entanto, pareceu-nos que a inclusão de um resumo e um conjunto de palavras chave em inglês para cada artigo seria uma concessão aceitável.

Sabemos que para conseguir este tipo de indexação também é importante a quantidade de citações de artigos publicados na revista. Outras revistas obrigam a que os artigos submetidos incluam citações para artigos da própria revista. Não nos parece que este tipo de obrigação seja decente, mas não podemos de deixar de pedir aos leitores e autores da Linguamática para, sempre que se adeque, façam citações a artigos publicados na revista.

Mas não será pela falta de indexação que nós, os editores, iremos baixar os braços.

Xavier Gómez Guinovart
José João Almeida
Alberto Simões

Comissão Científica

Alberto Álvarez Lugrís, Universidade de Vigo
Alberto Simões, Universidade do Minho
Aline Villavicencio, Universidade Federal do Rio Grande do Sul
Álvaro Iriarte Sanroman, Universidade do Minho
Ana Frankenberg-Garcia, ISLA e Universidade Nova de Lisboa
Anselmo Peñas, Universidad Nacional de Educación a Distancia
Antón Santamarina, Universidade de Santiago de Compostela
Antonio Moreno Sandoval, Universidad Autónoma de Madrid
António Teixeira, Universidade de Aveiro
Arantza Díaz de Ilarraza, Euskal Herriko Unibertsitatea
Belinda Maia, Universidade do Porto
Carmen García Mateo, Universidade de Vigo
Diana Santos, Linguateca/FCCN
Ferran Pla, Universitat Politècnica de València
Gael Harry Dias, Universidade Beira Interior
Gerardo Sierra, Universidad Nacional Autónoma de México
German Rigau, Euskal Herriko Unibertsitatea
Helena de Medeiros Caseli, Universidade Federal de São Carlos
Horacio Saggion, University of Sheffield
Iñaki Alegria, Euskal Herriko Unibertsitatea
Joaquim Llisterri, Universitat Autònoma de Barcelona
José Carlos Medeiros, Porto Editora
José João Almeida, Universidade do Minho
José Paulo Leal, Universidade do Porto
Joseba Abaitua, Universidad de Deusto
Juan-Manuel Torres-Moreno, Laboratoire Informatique d'Avignon - UAPV
Kepa Sarasola, Euskal Herriko Unibertsitatea
Lluís Padró, Universitat Politècnica de Catalunya
Maria das Graças Volpe Nunes, Universidade de São Paulo
Mercè Lorente Casafont, Universitat Pompeu Fabra
Mikel Forcada, Universitat d'Alacant
Patrícia Cunha França, Universidade do Minho
Pablo Gamallo Otero, Universidade de Santiago de Compostela
Rui Pedro Marques, Universidade de Lisboa
Salvador Climent Roca, Universitat Oberta de Catalunya
Susana Afonso Cavadas, University of Sheffield
Tony Berber Sardinha, Pontifícia Universidade Católica de São Paulo
Xavier Gómez Guinovart, Universidade de Vigo

Artigos de Investigação

Geocodificação de Documentos Textuais com Classificadores Hierárquicos Baseados em Modelos de Linguagem

Duarte Dias
IST
dcd@ist.utl.pt

Ivo Anastácio
INESC-ID Lisboa / IST
ivo.anastacio@ist.utl.pt

Bruno Martins
INESC-ID Lisboa / IST
bruno.g.martins@ist.utl.pt

Resumo

A maioria dos documentos textuais, produzidos no contexto das mais diversas aplicações, encontra-se relacionado com algum tipo de contexto geográfico. Contudo, os métodos tradicionais para a prospecção de informação em colecções de documentos vêem os textos como conjuntos de termos, ignorando outros aspectos. Mais recentemente, a recuperação de informação com suporte ao contexto geográfico tem capturado a atenção de diversos investigadores em áreas relacionadas com a prospecção de informação e o processamento de linguagem natural, envisionando o suporte para tarefas como a pesquisa e visualização de informação textual, com base em representações cartográficas. Neste trabalho, comparamos experimentalmente diferentes técnicas automáticas, as quais utilizam classificadores baseados em modelos de linguagem, para a atribuição de coordenadas geoespaciais de latitude e longitude a novos documentos, usando apenas o texto dos documentos como evidência de suporte. Medimos os resultados obtidos com modelos de linguagem baseados em n-gramas de caracteres ou de termos, usando colecções de artigos georreferenciados da Wikipédia em três línguas distintas, nomeadamente em Inglês, Espanhol e Português. Experimentamos também diferentes métodos de pós-processamento para atribuir as coordenadas geoespaciais com base nas classificações. O melhor método utiliza modelos de linguagem baseados em n-gramas de caracteres, em conjunto com uma técnica de pós-processamento que utiliza as coordenadas dos knn documentos mais similares, obtendo um erro de previsão médio de 265 Kilómetros, e um erro mediano de apenas 22 Kilómetros, para o caso da colecção da Wikipédia Inglesa. Para as colecções Portuguesa e Espanhola, as quais são significativamente mais pequenas, o mesmo método obteve um erro de previsão médio de 278 e 273 Kilómetros, respectivamente, e um erro de previsão mediano de 28 e de 45 Kilómetros.

Palavras chave

Processamento de Texto, Recuperação de Informação Geográfica, Geocodificação de Documentos

Abstract

Most text documents can be said to be related to some form of geographic context, although traditional text mining methods simply model documents as bags of tokens, ignoring other aspects of the encoded information. Recently, geographic information retrieval has captured the attention of many different researchers from fields related to text mining and data retrieval, envisioning the support for tasks such as map-based document indexing, retrieval and visualization. In this paper, we empirically compare automated techniques, based on language model classifiers, for assigning geospatial coordinates of latitude and longitude to previously unseen textual documents, using only the raw text of the documents as input evidence. We measured the results obtained with character-based or token-based language models over collections of georeferenced Wikipedia articles in four different languages, namelly English, Spanish and Portuguese. We also experimented with different post-processing methods for assigning the geospatial coordinates with basis on the resulting classifications. The best performing method combines character-based language models with a post-processing technique that uses the coordinates from the k most similar documents, obtaining an average prediction error of 265 Kilometers, and a median prediction error of just 22 Kilometers, in the case of the English Wikipedia collection. For the Spanish, and Portuguese collections, which are significantly smaller, the same method obtain an average prediction error of 273 and 278 Kilometers, respectively, and a median prediction error of 45 or 28 Kilometers.

Keywords

Text Mining; Geographic Information Retrieval; Document Geocoding;

1 Introdução

A maioria dos documentos textuais, produzidos no contexto das mais diversas aplicações, encontra-se relacionado com algum tipo de contexto geográfico. Recentemente, a Recuperação de Informação (RI) com base no contexto geográfico tem capturado a atenção de muitos investigadores em áreas relacio-

nadas com o processamento de língua natural e a prospecção de informação em grandes colecções de documentos textuais. Temos, por exemplo, que a tarefa de resolver referências a nomes de locais, apresentadas em documentos de texto, tem sido abordada em diversos trabalhos anteriores, com o objectivo de apoiar tarefas subsequentes em sistemas de RI geográficos, tais como a recuperação de documentos ou a visualização através de representações cartográficas (Lieberman e Samet, 2011). No entanto, a resolução de referências a nomes de locais apresenta vários desafios não-triviais (Leidner, 2007; Martins, Anastácio e Calado, 2010; Amitay et al., 2004), devido à ambiguidade inerente ao discurso em linguagem natural. Temos, por exemplo, que os nomes de locais são muitas vezes usados com outros significados não geográficos. Temos ainda que locais distintos são muitas vezes referidos pelo mesmo nome, ou que locais únicos são referidos por nomes diferentes. Além disso, existem muitos termos do vocabulário de uma dada língua, além dos nomes de locais, que podem surgir frequentemente associados com áreas geográficas específicas. Em lugar de tentar resolver corretamente as referências individuais a locais, que sejam apresentadas em documentos textuais, pode ser bastante interessante estudar métodos para a atribuição de âmbitos geográficos à totalidade dos conteúdos dos documentos (Wing e Baldridge, 2011; Adams e Janowicz, 2012). Os resultados poderão posteriormente ser aplicados em tarefas como a sumarização de colecções de documentos em representações baseadas em mapas (Bär e Hurni, 2011; Mehler et al., 2006; Erdmann, 2011).

Neste trabalho, é feita uma comparação de diferentes técnicas automáticas para a atribuição de coordenadas geoespaciais de latitude e longitude a novos documentos textuais, usando apenas o texto dos documentos como fonte de evidência, e utilizando uma representação discreta para superfície da Terra baseada numa decomposição em regiões triangulares de igual área. As diferentes regiões usadas na representação da Terra são inicialmente associadas aos documentos textuais que lhes pertencem (i.e., usamos todos os documentos presentes num conjunto de treino, que sejam conhecidos por se referir a cada uma das regiões em particular). De seguida, são construídas representações compactas (e.g., baseadas em modelos de linguagem suportados em n-gramas de caracteres ou de termos) a partir desses conjuntos de documentos georeferenciados, capturando as suas principais propriedades estatísticas. Novos documentos são então atribuídos à(s) região(ões) mais semelhante(s). Finalmente, são atribuídas as respectivas coordenadas geospaciais de latitude e longitude aos documentos, com base nas coordenadas centroides associadas à(s) região(ões). Foram ainda realizadas experiências com diferentes técnicas de pós-processamento para atribuir as coordenadas na etapa final, usando (i) as coordenadas centroide da região mais provável, (ii) uma média ponderada com as coordenadas das regiões mais prováveis, (iii) uma média ponderada com as coordenadas das regiões vizinhas da mais provável, e (iv) uma média ponderada com as coordenadas dos knn documentos de treino mais semelhantes (Shakhnarovich, Darrell e Indyk, 2006), que estejam contidos dentro da região mais provável para o documento.

Experiências com coleções de artigos da Wikipédia, contendo documentos em Inglês, Espanhol e Português, apresentaram bons resultados para a abordagem geral de geocodificação. O melhor método combina classificadores hierárquicos baseados em modelos de linguagem, usando n-gramas de caracteres, com a técnica de pós-processamento que utiliza a média ponderada das coordenadas dos knn documentos mais similares. Este método obteve um erro de previsão médio de 265 Kilómetros, e um erro de previsão mediano de apenas 22 Kilómetros, para o caso da colecção da Wikipédia Inglesa. Para as colecções Portuguesa e Espanhola, o mesmo método obteve um erro médio de 278 e 273 Kilómetros, e um erro mediano de 28 e 45 Kilómetros, respectivamente em cada uma das colecções.

O restante conteúdo deste artigo está organizado da seguinte forma: a Secção 2 apresenta trabalhos anteriores relacionados com a geocodificação de documentos. A Secção 3 apresenta a abordagem proposta, detalhando o uso dos classificadores baseados em modelos de linguagem, assim como as técnicas de pós-processamento propostas. A Secção 4 apresenta a validação experimental do método proposto, descrevendo os conjuntos de dados da Wikipédia que foram considerados, o protocolo experimental, e os resultados obtidos para as diferentes variações do método proposto. Finalmente, a Secção 5 sumariza as principais conclusões do trabalho, apontando ainda possíveis direções para trabalho futuro.

2 Trabalho Relacionado

A relação entre a linguagem e geografia tem sido um tema de interesse para os linguistas (Johnstone, 2010). Muitos estudos têm, por exemplo, mostrado que a geografia tem um impacto importante na relação entre termos do vocabulário e classes semânticas. Temos, por exemplo, que o termo *footboall*, nos Estados Unidos, se refere ao desporto em particular de futebol americano. No entanto, em regiões como a Europa, o termo *football* é geralmente associado a diferentes modalidades desportivas (e.g., o futebol ou, menos frequentemente, rugby). Termos como *praia* ou *neve* também são mais propensos a serem associados a determinados locais. Neste estudo,

estamos interessados em ver se os termos do vocabulário, e se conteúdos textuais no geral, podem ser usados para prever localizações geográficas.

Overell (2009) investigou o uso da Wikipédia como fonte de dados para a geocodificação de artigos textuais, assim como para a classificação de artigos por categorias, ou para a resolução de referências individuais a nomes de locais. O objetivo principal de Overell era a resolução de referências a locais em documentos, tarefa para a qual a geocodificação de documentos global pode servir como fonte de evidência. Para a geocodificação de documentos, Overell propôs um modelo simples que usa apenas os metadados disponíveis (e.g., título do artigo, hiperligações de entrada e saída para com outros documentos, etc.), e não o próprio texto dos documentos.

Adams e Janowicz (2012) estudaram a relação entre os tópicos em documentos textuais e a sua distribuição geoespacial. Enquanto que a maioria dos trabalhos anteriores, focados na extração de informação geográfica desde documentos, se baseiam em palavras-chave específicas, tais como os nomes de locais, Adams e Janowicz propuseram uma abordagem que usa apenas termos e expressões não geográficas, aferindo sobre se os termos textuais comuns são também bons na previsão de localizações geográficas. A técnica proposta usa o modelo *Lattent Dirichlet Allocation* (LDA) para descobrir tópicos latentes na coleção de documentos. LDA é essencialmente um método não-supervisionado que permite modelar o processo de geração de documentos através de misturas probabilísticas de tópicos, os quais são por sua vez modelados como distribuições de probabilidade sobre um vocabulário de termos. Depois de ajustar o modelo LDA a uma colecção de documentos, os autores utilizam a técnica *Kernel Density Estimation* (KDE) para interpolar uma superfície de densidade, correspondendo a uma região geoespacial, ao longo de cada tópico do modelo LDA. Notando que cada documento pode ser visto como uma mistura de tópicos, os autores utilizam operações de álgebra de mapas para combinar as superfícies de densidade geradas com base em cada tópico, finalmente atribuindo aos documentos o local geoespacial de maior densidade.

Eisenstein et al. (2010) investigaram as diferenças dialetais e as variações em interesses regionais nos utilizadores do Twitter, utilizando uma coleção de *tweets* georreferenciados e uma técnica baseada em modelos probabilísticos. Especificamente, estes autores tentaram georeferenciar os utilizadores do Twitter localizados nos Estados Unidos, com base nos conteúdos por sí produzidos. Eles concatenaram todos os *tweets* de cada utilizador distinto, e usaram distribuições Gaussianas para modelar as localizações dos utilizadores. As abordagens propostas no nosso artigo usam, alternativamente, uma representação discreta para a superfície da Terra, em conjunto com modelos probabilísticos mais simples construídos sobre essa representação discreta.

Anastácio, Martins e Calado (2010) estudaram abordagens heurísticas para atribuir âmbitos geográficos a documentos textuais, com base no reconhecimento de referências a locais nos documentos, posteriormente combinando as referências reconhecidas. Os autores compararam especificamente abordagens com base (i) na frequência de ocorrência associada às referências a locais, (ii) na sobreposição geoespacial entre caixas delimitadoras associadas às referências, (iii) na distância hierárquica entre as referências, usando uma taxonomia geográfica de divisões administrativas, e (iv) na propagação de informação sobre um grafo codificando relações entre locais, usando novamente uma taxonomia geográfica com divisões administrativas. Experiências com uma coleção de páginas Web do *Open Directory Project*[1] mostraram que a técnica baseada na distância hierárquica consegue bons resultados. Neste trabalho, estamos também a estudar abordagens para a geocodificação do conteúdo de documentos textuais, mas neste caso usando directamente o texto como fonte de evidência, em alternativa à utilização de referências a locais nos textos.

Wing e Baldridge (2011), num estudo muito semelhante ao que é relatado no presente artigo, compararam abordagens diferentes para a geocodificação automática de documentos, usando também como base modelos estatísticos derivados de um conjunto vasto de documentos já geocodificados, como a Wikipédia. Os autores utilizaram a divergência de Kullback-Leibler entre um modelo de linguagem construído sobre um documento de teste, e modelos de linguagem para cada célula de uma representação discreta para a superfície da Terra, como forma de prever a célula mais provável de conter o documento de teste. Uma abordagem semelhante foi posteriormente proposta para a resolução temporal de documentos, sendo capaz de determinar a data da publicação de um dado artigo, com base no texto (Kumar, Lease e Baldridge, 2011). Novamente neste trabalho, os autores construíram histogramas que codificam a probabilidade de diferentes períodos temporais para um documento, mais tarde usando a divergência de Kullback-Leibler para fazer as previsões. O trabalho relatado neste artigo é muito semelhante ao de Wing e Baldridge, mas nós propomos utilizar (i) um esquema diferente para particionar o conjunto de documentos em regiões de igual área, de acordo com sua localização geoespacial, (ii) uma abordagem diferente para a classificação de documentos através de modelos de linguagem, (iii) uma abordagem de

[1] `http://www.dmoz.org/`

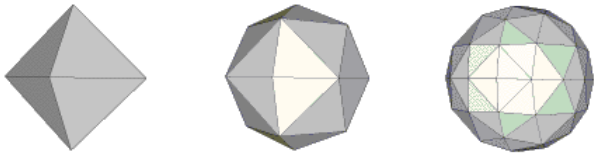


Figura 1: Decomposição da superfície terrestre com grelhas triangulares de resolução zero, um e dois.

decomposição hierárquica para melhorar o desempenho computacional do método de classificação, e (iv) diferentes técnicas de pós-processamento para atribuir as coordenadas geoespaciais com base na classificação obtida, i.e. com base nas pontuações associadas a cada célula da decomposição da Terra.

3 Geocodificação de Documentos

A abordagem de geocodificação de documentos textuais, proposta neste artigo, baseia-se na discretização da superfície da Terra num conjunto de células triangulares, o que nos permite prever os locais, a associar aos documentos, com abordagens estatísticas padrão para a modelação de atributos discretos. No entanto, ao contrário de autores anteriores como Serdyukov, Murdock e van Zwol (2009) ou Wing e Baldridge (2011), os quais utilizaram uma grelha de células rectangulares, nós utilizamos uma grelha triangular, obtida através de um método de decomposição da superfície da Terra conhecido pela designação de *Hierarchical Triangular Mesh*[2] (Dutton, 1996; Szalay et al., 2005). Esta estratégia resulta numa grelha triangular que preserva uma área aproximadamente igual para cada célula, em lugar de resultar em células de tamanho variável, com regiões que se encolhem de acordo com a latitude, tornando-se progressivamente menores e alongadas à medida que se aproximam dos polos. Importa aqui referir que a nossa representação ignora todas as regiões geográficas de nível semanticamente superior, como os estados, países ou continentes. No entanto, esta representação é apropriada para o propósito de geocodificar documentos textuais, uma vez que os mesmos podem estar relacionados com regiões geográficas que não se encaixam numa divisão administrativa da superfície da Terra.

A *Hierarchical Triangular Mesh* (HTM) oferece uma decomposição multi-nível recursiva para uma aproximação esférica da superfície da Terra – ver as

[2] `http://www.skyserver.org/htm/Old_default.aspx`

Figura 2: Decomposição de triângulos esféricos.

Figuras 1 e 2, ambas adaptadas de imagens originais do sítio Web descrevendo a abordagem HTM. A decomposição começa num nível zero com um octaedro e, projectando as arestas do octaedro sobre a esfera terrestre, criam-se 8 triângulos esféricos (i.e., triângulos projectados sobre a esfera terrestre), 4 no hemisfério Norte e 4 sobre o hemisfério Sul. Quatro destes triângulos partilham um vértice nos polos, e os lados opostos formam o Equador. Cada um dos 8 triângulos esféricos pode ser dividido em quatro triângulos menores, através da introdução de novos vértices nos pontos médios de cada aresta, e adicionando um segmento de arco de grande círculo para conectar os novos vértices. Este processo de subdivisão pode repetir-se recursivamente, até chegarmos ao nível de resolução desejado. Os triângulos nesta grelha são as células utilizadas na nossa representação da Terra, e cada triângulo, em qualquer resolução, é representado por um identificador único. Para cada localização dada por um par de coordenadas sobre a superfície da esfera terrestre, existe um identificador que representa o triângulo, considerando uma resolução em particular, que contém o ponto correspondente.

Note-se que o mecanismo de representação proposto contém um parâmetro k que controla a resolução, i.e., a área das células. Na nossa aplicação de classificação de documentos, a utilização de células de granularidade elevada pode levar a estimativas muito grosseiras, muito embora a precisão da classificação, com uma resolução mais fina, também possa diminuir substancialmente, devido a termos dados insuficientes para construir os modelos de linguagem associados a cada célula. Nas nossas experiências, o parâmetro k variou entre os valores de 4 e 10, com 0 correspondendo a uma divisão de primeiro nível. A Tabela 1 apresenta o número de células gerado em cada um dos níveis de resolução considerados. O número de células n para uma resolução k é dado por $n = 8 * 4^k$. A Tabela 1 mostra também a área, em Kilómetros quadrados, correspondente a cada célula da representação.

Com base na representação em células para a superfície da Terra, dada pelo método HTM, foi depois utilizado o software *Alias-I LingPipe*[3] para construir modelos de linguagem baseados em n-gramas de caracteres ou de termos, usando-se segui-

[3] `http://alias-i.com/lingpipe`

Resolução	4	6	8	10
Número total de células	2,048	32,768	524,288	2,097,152
Área apróximada de cada célula (km^2)	28,774.215	17,157.570	1,041.710	261.675

Tabela 1: Número de células e a sua área aproximada em grelhas triangulares de diferentes resoluções.

damente estes modelos para associar, a cada célula da representação, a probabilidade de a mesma ser a melhor classe para um novo documento textual.

De forma resumida, temos que os classificadores desenvolvidos com o ***LingPipe*** tomam as suas decisões com base na probabilidade conjunta de documentos textuais e categorias, usando modelos de linguagem baseados em n-gramas de caracteres ou de termos textuais (i.e., nas nossas experiências, testamos estas duas abordagens diferentes de classificação). A ideia geral envolve estimar uma probabilidade $P(txt|cat)$ para cada categoria cat, estimar uma distribuição multinomial $P(cat)$ sobre as categorias e calcular o logaritmo das probabilidades conjuntas para as categorias e documentos, de acordo com regra de Bayes, produzindo-se assim:

$$\log_2 P(cat, txt) \propto \log_2 P(txt|cat) + \log_2 P(cat) \quad (1)$$

Na fórmula, $P(txt|cat)$ é a probabilidade de ver um determinado texto txt no modelo de linguagem para a categoria cat, e $P(cat)$ é a probabilidade marginal atribuída pela distribuição multinomial sobre as categorias. O livro de Carpenter e Baldwin (2011) apresenta mais detalhes sobre os modelos de linguagem usados para estimar $P(txt|cat)$, e sobre a distribuição multinomial $P(cat)$ sobre as categorias (ou seja, sobre as células da nossa representação da Terra). Esta última multinomial é basicamente estimada utilizando o critério MAP (i.e., *maximum a posteriori probability*) com hipóteses *a priori* aditivas (i.e., *priors* de Dirichlet).

No que diz respeito aos modelos de linguagem baseados em n-gramas de caracteres, temos essencialmente modelos de linguagem generativos com base na regra da cadeia, em que as estimativas são suavizadas através de interpolação linear com modelos de ordem inferior, e onde há uma probabilidade de 1.0 para a soma das probabilidades de todas as sequências de um comprimento especificado. Os nossos modelos baseados em n-gramas de caracteres consideram sequências de 8 caracteres. Quanto aos modelos de linguagem baseados em termos, são capturadas as sequências de termos com um modelo de bi-gramas, e modelados os espaços em branco e os símbolos desconhecidos separadamente. A segmentação dos textos em termos é feita através do um método disponível no software ***LingPipe***, o qual utiliza regras comuns a diferentes línguas indo-europeias, semelhantes às regras consideradas no MUC-6[4]. Um termo é assim definido como uma sequência de caracteres satisfazendo um dos seguintes padrões, enquanto que os espaços em branco (i.e., os separadores entre termos) correspondem a sequências de símbolos onde se incluem os espaços, tabulações e mudanças de linha:

- Termos alfa-numéricos. i.e. sequências de letras ou de dígitos;
- Termos numéricos. i.e. sequências de números, virgulas, e pontos;
- Hífens, i.e. sequências de um ou mais hífens;
- Igualdades, i.e. sequências de um ou mais símbolos de igualdade;
- Duplas-aspas, i.e. diferentes formas de representar duplas-aspas nos textos.

O leitor pode consultar o livro de Carpenter e Baldwin (2011) para obter informações mais detalhadas sobre o método de classificação que é aqui usado.

Depois de termos probabilidades atribuídas a cada uma das células na nossa representação da Terra, calculamos as coordenadas geoespaciais de latitude e longitude, com base nas coordenadas centroide para a(s) célula(s) mais provável(eis). Nesta fase em particular, testámos quatro diferentes técnicas de pós-processamento dos resultados:

1. Atribuir coordenadas geoespaciais com base no centroide da célula mais provável.
2. Atribuir coordenadas geoespaciais de acordo com uma média ponderada das coordenadas centroide para todas as células possíveis, em que os pesos são as probabilidades atribuídas a cada uma das células pelo classificador.
3. Atribuir coordenadas geoespaciais de acordo com uma média ponderada das coordenadas centroide para a célula mais provável e para as suas vizinhas adjacentes na grelha triangular, novamente usando como pesos as probabilidades atribuídas a cada uma das células.

[4] `http://cs.nyu.edu/faculty/grishman/muc6.html`

4. Atribuir coordenadas geoespaciais de acordo com uma média ponderada das coordenadas associadas aos knn documentos mais semelhantes nos dados de treino, filtrados de acordo com a pertença das suas coordenadas à célula mais provável descoberta pelo classificador.

Os métodos dois e três da enumeração anterior exigem que o classificador retorne probabilidades bem calibradas sob as classes possíveis, enquanto que a abordagem baseada em modelos de linguagem, utilizada nas nossas experiências, é conhecida por produzir estimativas de probabilidade distorcidas e muito extremas. Na literatura de aprendizagem automática, existem muitos métodos para calibrar as probabilidades retornadas por métodos de classificação, mas a maioria desses métodos são definidos apenas para problemas de classificação binários (Gebel e Weihs, 2007). No nosso problema particular de classificação multi-classe, optamos por processar os valores retornados pelos classificadores baseados em modelos de linguagem através de uma função sigmoide da forma $(\sigma \times score)/(\sigma - score + 1)$, onde o parâmetro σ que controla o gradiente da curva foi ajustado empiricamente.

No que diz respeito ao quarto método de pós-processamento, nós medimos a semelhança entre os documentos de acordo com semelhança do cosseno, entre os vectores de características que os representam. As características correspondem à frequência de ocorrência de uni-gramas de termos. Nas nossas experiências, variou-se o parâmetro knn entre os valores de cinco e vinte documentos.

Embora os classificadores baseados em modelos de linguagem possam ser usados directamente para atribuir documentos às células mais prováveis, eles na prática são muito ineficientes quando se considera uma resolução fina, devido ao número elevado de classes – ver a Tabela 1 – e devido à necessidade de estimar, para cada documento, a sua probabilidade de ter sido gerado pelo modelo de linguagem correspondente a cada classe. Neste trabalho, propomos usar uma abordagem de classificação hierárquica, onde em vez de um classificador único considerando todas as células de uma grelha triangular detalhada, codificando a superfície da Terra, usamos uma hierarquia de classificadores com dois níveis. O primeiro nível corresponde a um modelo único de classificação utilizando células geradas com uma divisão grosseira da superfície terrestre. O segundo nível corresponde a classificadores diferentes, um para cada classe do primeiro nível, codificando diferentes partes da Terra com uma resolução mais elevada. Com este esquema hierárquico, a classificação pode ser feita com muito mais eficiência, uma vez que os documentos precisam de ser avaliados com menos modelos de linguagem. Ainda no que diz respeito a classificação hierárquica, nós também tiramos partido das propriedades da técnica HTM, de modo a reduzir o número de classes em cada um dos modelos gerados no último nível de hierarquia de classificação. Recursivamente, verificamos se uma dada célula não contém quaisquer documentos de treino atribuídos na resolução actualmente considerada, e se apenas uma das células vizinhas na grelha triangular contém documentos. Nestes casos, usamos uma única classe com base na grelha triangular com a resolução imediatamente menor, como forma de representar a região no modelo de classificação.

Num trabalho relacionado anterior focado na língua Inglesa, Wing e Baldridge (2011) relataram resultados muito precisos (i.e., um erro de previsão mediano de apenas 11.8 km, e um erro médio de 221 km), com uma abordagem de classificação semelhante, embora não hierárquica, baseada na divergência de Kullback-Leibler entre modelos de linguagem. No entanto, estes autores também afirmam que uma execução completa de todas as suas experiências (i.e., seis estratégias diferentes) necessitou de cerca de 4 meses em tempo de computação num processador Intel Xeon E5540 de 64-bit, utilizando cerca de 10-16 GB de RAM. A abordagem de classificação hierárquica permite reduzir substancialmente o esforço computacional exigido, tendo-se que as nossas experiências se realizaram em hardware semelhante durante apenas alguns dias.

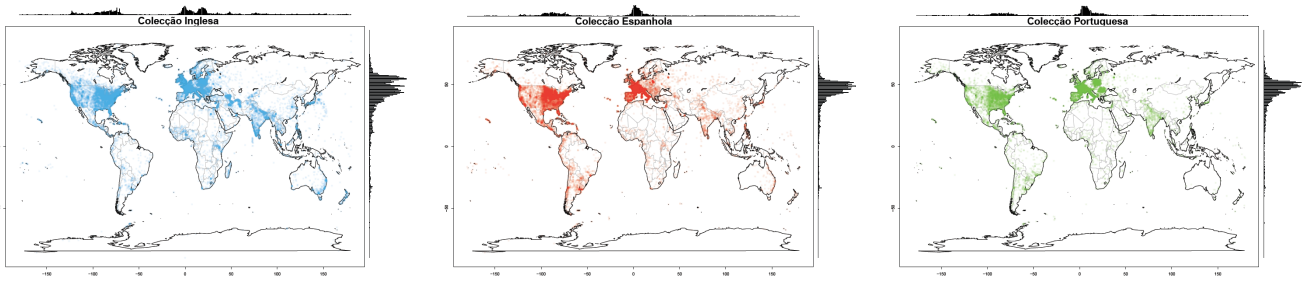
4 Avaliação Experimental

Vamos agora descrever a metodologia de avaliação experimental utilizada para comparar os métodos propostos, discutindo depois os resultados obtidos. Nas experiências aqui relatadas, foram utilizados artigos textuais das versões Inglesa, Espanhola e Portuguesa da Wikipédia, extraídas de *dumps* produzidos em 2012 (i.e., os *dumps* de 2012-06-01 no caso das Wikipédias Inglesa e Portuguesa, e o *dump* de 2012-05-15 no caso da Wikipédia Espanhola). Incluem-se nestas amostras um total de 393,294, 119,572 e 96,643 artigos, respectivamente em Inglês, Espanhol e Português, os quais se encontram associados a coordenadas de latitude e longitude. Estudos anteriores já demostraram que os artigos da Wikipédia são uma fonte adequada de conteúdos textuais georreferenciados para este tipo de testes (Overell, 2009; Wing e Baldridge, 2011).

Temos especificamente que foram processados todos os documentos dos *dumps* da Wikipédia usando o software *dmir-wiki-parser*[5], por forma a extrair o texto dos artigos, e por forma a extrair também as coordenadas geo-espaciais, usando padrões manual-

[5] `http://code.google.com/p/dmir-wiki-parser/`

Figura 3: Mapas tematicos representativos das distribuições geográficas dos documentos da Wikipédia.



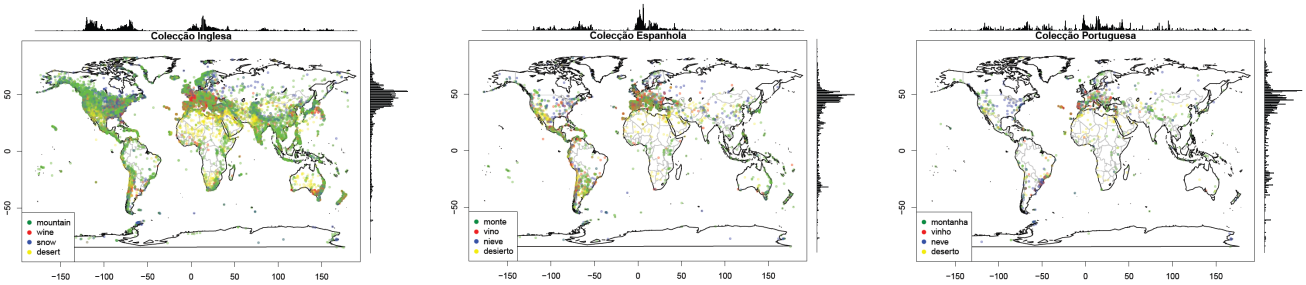
mente definidos para capturar alguns dos múltiplos modelos e formatos usados para expressar latitude e longitude na Wikipédia (i.e., valores situados na *infobox* de cada página). Considerando uma ordem aleatória para os artigos processados desta forma, cerca de 90% dos artigos em cada língua foram utilizados no treino de modelos de classificação (i.e., um total de 353,294 artigos em Inglês, 89,572 em Espanhol, e 77,314 em Português) e os outros 10% foram utilizados para a validação dos métodos propostos (isto é, um total de 40,000, 30,000 e 19,329 artigos, respectivamente em Inglês, Espanhol e Português). A Tabela 2 apresenta uma caracterização estatística para os conjuntos de documentos considerados, enquanto que a Figura 3 ilustra a distribuição geoespacial dos locais associados com os documentos nas três diferentes colecções. Pode-se observar que algumas regiões geográficas (por exemplo a América do Norte ou a Europa) são consideravelmente mais densas em termos de associações a documentos do que outras (por exemplo, África). Verificamos ainda que na colecção Portuguesa existe uma maior concentração de artigos na Europa e na América do Sul (i.e., no Brasil), e que na colecção Espanhola existe uma maior concentração de artigos na Europa e nos países latinos da América do Sul. Além disso, temos que os oceanos e outras grandes massas de água são escassos em associações a documentos da Wikipédia. Isto implica que o número de classes que têm de ser consideradas nos nossos modelos de classificação é muito menor do que os números teóricos de classes apresentados na Tabela 1. No nosso conjunto de dados em Inglês, existe um total de 1,123 células contendo associações para documentos numa resolução de nível 4, e um total de 8,320, 42,331 e 144,693 células, respectivamente quando considerando resoluções de 6, 8 e 10. Estes números são significativamente inferiores no caso das colecções em Espanhol e Português, com apenas 928 e 886 células contendo associações para documentos numa resolução de nível 4, respectivamente no caso das colecções Espanhola e Portuguesa. Ver Tabela 4 onde se apresenta o número de células diferentes em cada colecção, junto com o número médio de documentos de treino por cada célula.

Importa referir que a associação de coordenadas geoespaciais aos documentos das Wikipédias Portuguesa e Espanhola levantou alguns problemas, dado que apenas um número de reduzido destas páginas contém menções explícitas a coordenadas nas suas *infoboxes*. Como forma de contornar esta limitação, utilizámos os links existentes entre as páginas nas várias línguas da Wikipédia, associando assim as coordenadas geoespaciais existentes para as páginas da Wikipédia Inglesa, às páginas equivalentes nas versões Portuguesa e Espanhola. Temos assim que muitos dos documentos usados nas 3 línguas diferentes se referem na prática aos mesmos conceitos e entidades do mundo real. Especificamente nas colecções referentes às Wikipédias Portuguesa e Espanhola, e no total dos documentos usados para treino e teste, temos respectivamente que 62,973 e 81,181 dos documentos são referentes a conceitos também existentes na colecção da Wikipédia Inglesa (i.e., as coordenadas geoespaciais são exactamente iguais, muito embora as descrições textuais sejam diferentes). No total, temos que 50,322 dos documentos considerados são referentes a conceitos partilhados entre as três colecções da Wikipédia.

Como forma inicial de validar a hipótese de que os termos textuais podem ser indicativos de localizações geográficas específicas, filtramos primeiro os documentos de acordo com a ocorrência de termos particulares. De seguida, representámos esses documentos num mapa. A Figura 4 mostra a incidência geográfica de termos textuais diferentes nas suas traduções para as três linguas consideradas, nomeadamente os termos, *montanha*, *vinho*, *neve*, e *deserto*. As figuras mostram que estes termos particulares são mais associados às regiões que seriam esperadas (i.e., termos como *vinho* estão mais associados a regiões como França, ou termos como *deserto* estão mais associados ao Norte de África).

Usando os três conjuntos de documentos da Wikipédia, fizemos experiências com modelos de classificação considerando diferentes níveis de resolução para as células. A Tabela 3 apresenta os resultados obtidos para alguns dos diferentes métodos em estudo (ou seja, para os dois tipos de classifica-

Figura 4: Mapas tematicos representativos das distribuições geográficas de certos termos.



Wikipédia EN	Treino	Teste
Num. Documentos	390,032	40,000
Num. Termos	160,508,876	16,696,639
Média Termos/Doc.	411	417
St.Dev. Termos/Doc.	875.517	901.215

Wikipédia ES	Treino	Teste
Num. Documentos	89,572	30,000
Num. Termos	29,633,769	9,788,169
Média Termos/Doc.	330	326
St.Dev. Termos/Doc.	1016.289	960.214

Wikipédia PT	Treino	Teste
Num. Documentos	77,314	19,329
Num. Termos	13,897,992	3,433,134
Média Termos/Doc.	179	179
St.Dev. Termos/Doc.	615.192	615.250

Tabela 2: Caracterização dos conjuntos de dados da Wikipédia usados na avaliação experimental.

dores, e considerando as três primeiras estratégias de pós-processamento, em que não se usam documentos similares), mostrando os valores de erro para cada tamanho de célula. Os erros de previsão apresentados na Tabela 3 correspondem à distância em Kilómetros, calculada através das formulas de Vincenty[6], com base nas coordenadas estimadas e nas coordenadas indicadas na Wikipédia. Os valores de exactidão correspondem ao rácio entre o número de classificações correctas (ou seja, aquelas onde a célula mais provável contém as verdadeiras coordenadas geoespaciais de latitude e longitude, tal como associadas ao documento) e o número de classificações efectuado. Os valores $k1$ e $k2$ correspondem à resolução usada na representação da Terra, para cada nível do classificador hierárquico.

Os valores da Tabela 3 mostram que o método de classificação proposto obtém melhores resultados com o aumento do número de documentos de treino, tendo-se que os resultados são um pouco melhores no caso da colecção em Inglês, em comparação com os resultados para as colecções em Espanhol e Português. O método correspondente ao uso de modelos de linguagem baseados em n-gramas de caracteres, utilizando uma resolução do segundo nível de 8 (i.e., áreas de classificação de 1041 Km^2) obteve os melhores resultados, com uma exactidão de cerca de 0.4 na tarefa de encontrar a célula correta, no caso da colecção em Inglês, enquanto que a atribuição de coordenadas geoespaciais aos documentos teve um erro de 268 Kilómetros, em média, também no caso da colecção em Inglês. No caso concreto desse teste, os documentos que foram atribuídos à célula correta foram associados a coordenadas que se encontravam a uma distância média de 14 Kilómeteos para com as coordenadas corretas. Podemos também observar que para uma resolução 10 (i.e., para uma área de classificação de 262 Km^2), os resultados pioram substancialmente, provavelmente devido ao reduzido número de documentos de treino associado a cada célula do modelo, como demonstrado na Tabela 4. Os resultados da Tabela 3 mostram ainda que tanto a segunda como a terceira técnica de pós-processamento melhoram geralmente os resultados da geocodificação sobre o método base em que se atribuem as coordenadas do ponto centróide da célula mais provável. No entanto, os resultados mostram apenas uma ligeira melhoria para esta técnica, e acreditamos que isto se deve ao facto dos nossos classificadores, baseados em modelos de linguagem, não fornecerem estimativas de probabilidade precisas e bem calibradas, tendo-se que a nossa técnica de calibração baseada num pós-processamento dos valores, através de uma função sigmoide, continua a produzir resultados demasiado extremos.

A Tabela 5 apresenta os resultados obtidos com modelos de linguagem baseados em n-gramas de caracteres (ou seja, com o melhor método de acordo com a experiência anterior), quando se utiliza o quarto método de pós-processamento, em que se atribuem coordenadas de latitude e longitude através do ponto centroide das coordenadas associadas aos knn documentos mais semelhantes, contidos dentro da célula mais provável para cada documento. A pri-

[6] `http://en.wikipedia.org/wiki/Vincenty`

Método	Resolução k1	Resolução k2	Exactidão do Classificador 1º Nível	Exactidão do Classificador 2º Nível	Distância Geoespacial – Centroide Média	Centroide Mediana	Todas as Células Média	Todas as Células Mediana	Células Vizinhas Média	Células Vizinhas Mediana
Documentos da Wikipédia Inglesa										
N-gramas caracteres	0	4	**0.9609**	**0.8354**	405.214	240.017	438.762	228.829	386.271	219.379
	1	6	0.9411	0.6669	**254.846**	62.846	282.874	71.119	257.741	65.551
	2	8	0.9283	0.3989	268.480	**25.757**	283.761	48.039	269.493	28.569
	3	10	0.8912	0.1615	281.669	30.405	287.909	51.595	281.755	30.464
N-gramas termos	0	4	0.9444	0.5209	693.764	240.017	544.543	228.829	691.899	219.379
	1	6	0.9103	0.4244	433.224	92.770	729.546	303.055	444.766	120.561
	2	8	0.8909	0.2747	455.025	44.621	572.406	191.522	457.858	50.349
	3	10	0.8297	0.1305	547.760	42.162	591.111	123.457	547.996	41.982
Documentos da Wikipédia Espanhola										
N-gramas caracteres	0	4	**0.9526**	**0.7725**	405.379	233.200	403.748	232.134	404.222	232.099
	1	6	0.9162	0.5890	281.810	68.098	280.973	67.570	281.616	67.887
	2	8	0.9015	0.3339	280.293	30.629	**279.639**	**30.293**	280.241	30.480
	3	10	0.8495	0.0948	327.278	43.311	326.754	43.122	327.272	43.311
N-gramas termos	0	4	0.8878	0.6450	729.810	263.677	729.015	262.576	729.158	262.575
	1	6	0.8249	0.4867	559.329	82.117	558.906	81.726	559.198	81.940
	2	8	0.8049	0.3015	570.181	39.035	569.633	38.429	570.112	38.824
	3	10	0.7379	0.1082	692.899	36.376	692.416	35.611	692.874	36.317
Documentos da Wikipédia Portuguesa										
N-gramas caracteres	0	4	**0.9470**	**0.7680**	381.253	231.011	378.907	229.360	379.165	229.378
	1	6	0.8966	0.5172	264.155	74.424	**262.699**	73.361	263.714	73.918
	2	8	0.8768	0.2392	275.398	46.781	274.166	**45.917**	275.340	46.533
	3	10	0.8303	0.0748	286.686	52.098	286.050	51.472	286.681	52.057
N-gramas termos	0	4	0.8863	0.6674	584.522	252.107	582.190	250.372	582.847	250.485
	1	6	0.8164	0.4457	460.500	86.201	459.485	85.350	460.140	85.593
	2	8	0.8003	0.2317	454.941	50.158	453.643	48.862	454.853	50.032
	3	10	0.7438	0.0886	491.483	48.462	490.463	46.906	491.469	48.457

Tabela 3: Resultados obtidos na geocodificação de documentos com diferentes classificadores.

	Resolução HTM			
Wikipédia EN	4	6	8	10
Num. Células	1,107	8,404	42,907	146,498
Média Docs/Célula	352	46	9	3

	Resolução HTM			
Wikipédia ES	4	6	8	10
Num. Células	928	5,105	18,655	52,348
Média Docs/Célula	97	18	5	2

	Resolução HTM			
Wikipédia PT	4	6	8	10
Num. Células	886	4,638	17,772	48,692
Média Docs/Célula	96	18	5	2

Tabela 4: Número de documentos de treino disponível para cada célula do método HTM.

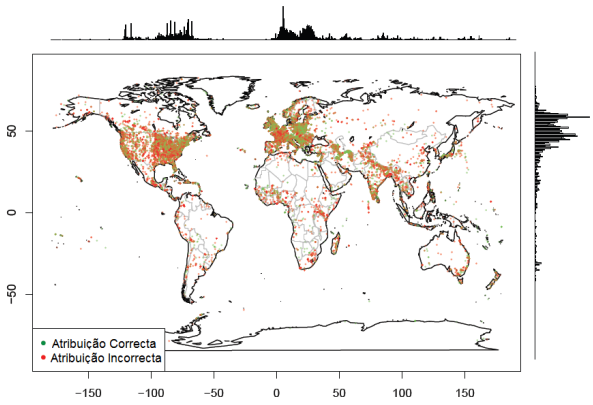


Figura 5: Localizações estimadas para os documentos de teste da Wikipédia Inglesa.

meira coluna da Tabela 5 indica o número de documentos semelhantes que foi considerado, enquanto que os valores de $k2$ correspondem à resolução para a representação da Terra, utilizada no segundo nível do classificador hierárquico.

A melhor configuração para as três linguas corresponde a uma resolução de 8. Os resultados mostram ainda que usando os 5 documentos mais similares se obtêm melhores resultados para as colecções Inglesa e Espanhola, com uma distância média de 265 e 278 Kilómetros, respectivamente, e uma distância mediana de 22 e 29 Kilómetros, respectivamente. Na colecção Portuguesa obtêm-se melhores resultados usando os 10 documents mais similares, com uma distância média de 273 Kilómetros e uma distância mediana de 45 Kilómetros.

Uma visualização dos resultados obtidos com o melhor método de geocodificação encontra-se apresentada na Figura 5, em que o mapa representa a distribuição geoespacial das localizações previstas para os documentos da colecção em Inglês. A figura mostra que os erros são uniformemente distribuídos, e mostra ainda que a Europa e a América do Norte

knn	k1	k2	Inglês Média	Inglês Mediana	Espanhol Média	Espanhol Mediana	Português Média	Português Mediana
5	0	4	289.750	90.515	290.021	77.583	260.805	76.665
	1	6	235.702	34.005	260.862	34.695	244.363	46.324
	2	8	265.734	**22.315**	277.895	**27.865**	273.218	44.612
	3	10	281.442	30.209	327.198	43.273	286.649	51.972
10	0	4	274.515	68.252	279.100	60.258	249.039	58.964
	1	6	233.982	32.092	**260.049**	33.824	**243.443**	44.839
	2	8	265.655	22.371	278.205	28.266	273.355	**44.587**
	3	10	281.460	30.208	327.215	43.324	286.685	51.960
15	0	4	271.045	64.008	277.652	59.384	247.373	55.599
	1	6	**233.928**	32.243	260.666	35.257	243.880	45.133
	2	8	265.744	22.480	278.511	28.723	273.600	44.896
	3	10	281.464	30.172	327.211	43.298	286.689	51.961
20	0	4	270.337	63.373	278.069	60.767	247.886	55.217
	1	6	234.197	32.687	261.367	36.155	244.437	45.485
	2	8	265.869	22.640	278.734	28.899	273.797	45.109
	3	10	281.466	30.170	327.213	43.298	286.689	51.961

Tabela 5: Resultados obtidos na geocodificação de documentos com o método de pós-processamento baseado na utilização dos knn documentos de treino mais similares.

continuam a ser as regiões de maior densidade.

A Figura 6 ilustra a distribuição para os erros produzidos pelos classificadores baseados em n-gramas de caracteres, em termos da distância entre as coordenadas estimadas e as coordenadas verdadeiras, quando se utiliza o método de pós-processamento considerado como *baseline*, e o método que utiliza as coordenadas dos knn documentos mais similares, para as três linguas. Estes gráficos apresentam o número de documentos em que o erro (i.e., a distância) é maior ou igual do que um dado valor, utilizando eixos logarítmicos. A Figura 6 mostra que o método de pós-processamento com base na análise dos documentos mais semelhantes atribui coordenadas à maioria dos exemplos com um pequeno erro em termos de distância, e com apenas cerca de 100 documentos correspondendo a um erro maior do que 10,000 Kilómetros, no caso da colecção em Inglês. Piores resultados são apresentados para o método base, com cerca de 200 documentos em que se observa um erro maior do que 10,000 Kilómetros nas coordenadas previstas, mais uma vez no caso da colecção de Wikipédia em Inglesa.

Finalmente, na Tabela 6 sumarizam-se os resultados para a melhor configuração em cada lingua, os quais foram sempre obtidos com classificadores baseados em n-gramas de caractéres. A Tabela 6 apresenta ainda os valores correspondentes a um intervalo de confiança de 95% para os erros médios e medianos obtidos com a melhor configuração.

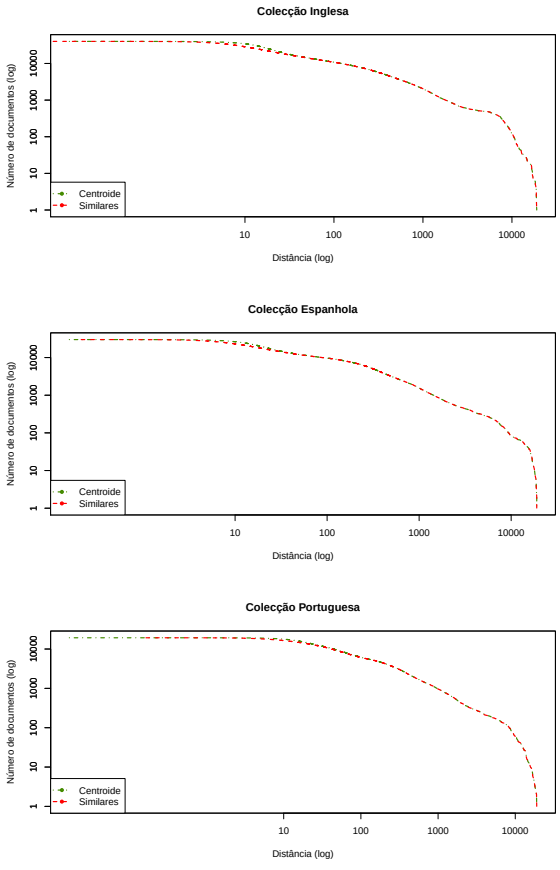


Figura 6: Distribuição dos valores de erro obtidos, em termos da distância geoepacial para com as coordenadas de latitude e longitude correctas.

5 Conclusões e Trabalho Futuro

Este trabalho avaliou diferentes métodos para a geocodificação de documentos textuais, os quais uti-

	Resolução			Distância			
Língua	k1	k2	knn	Média	Média (95%)	Mediana	Mediana (95%)
Inglês	2	8	5	265.734	255.230 - 276.237	22.315	21.859 - 22.771
Espanhol	2	8	5	277.895	265.725 - 290.065	27.865	26.996 - 28.734
Português	2	8	10	273.355	258.838 - 287.871	44.587	43.498 - 45.676

Tabela 6: Sumarização dos melhores resultados obtidos.

lizam classificadores baseados em modelos de linguagem para a atribuição de regiões geo-espaciais aos documentos, fazendo ainda um pós-processamento dos resultados da classificação por forma a atribuir as coordenadas geoespaciais de latitude e longitude mais prováveis. Mostramos que a identificação automática das coordenadas geoespaciais de um documento, com base apenas no seu texto, pode ser feita com alta precisão, utilizando métodos simples de classificação supervisionada, e usando uma representação discreta para a superfície da Terra, com base numa grelha triangular hierárquica. O método proposto é simples de implementar, e tanto o treino como os testes podem ser facilmente paralelizados por forma a processar grandes colecções de documentos. A nossa estratégia de geocodificação mais eficaz utiliza modelos de linguagem baseados em n-gramas de caracteres, e atribui as coordenadas de latitude e longitude através do centroide das coordenadas dos knn documentos de treino mais semelhantes ao documento sob análise, contidos dentro da região mais provável para cada documento.

Importa referir que as experiências relatadas neste artigo foram efectuadas separadamente com três colecções de documentos distintas, nomeadamente em Inglês, Espanhol, e Português. Para trabalho futuro, seria interessante realizar experiências com colecções de documentos ainda noutras linguas, e seria também interessante introduzir um terceiro nível de classificação no método proposto, por forma a que os documentos fossem inicialmente classificados de acordo com a língua, e posteriormente processados com os modelos de geocodificação correspondentes. Assim, e usando por exemplo os dados das várias Wikipédias, seria possível construir um sistema que automáticamente geocodificasse documentos, independentemente da sua língua.

Existem muitas aplicações possíveis para o método de geocodificação de documentos descrito no presente documento. Uma aplicação em particular, que estamos a considerar num trabalho em curso, relaciona-se com o uso das distribuições de probabilidade sobre as células, da nossa representação da Terra, na construção de mapas temáticos que mostrem a incidência geográfica de determinadas construções extraídas dos textos (por exemplo, mapas mostrando a distribuição geográfica das opiniões expressas em relação a determinados temas). No entanto, importa reforçar que a abordagem de classificação proposta, com base em modelos de linguagem, não fornece estimativas de probabilidade precisas e bem calibradas para as diferentes classes envolvidas no problema, focando-se apenas na tarefa mais simples de prever qual a classe mais provável. Para trabalho futuro, em lugar de usarmos um método heurístico de calibração com base no pós-processamento dos valores retornados pelo classificador, gostaríamos de experimentar outras abordagens de classificação para a atribuição da(s) célula(s) mais provável aos documentos, tais como por exemplo modelos de máxima entropia (i.e., regressão logística). Também gostaríamos de experimentar com modelos de máxima entropia usando restrições nas expectativas especificando afinidades entre os termos e as classes (Druck, Mann e McCallum, 2008), ou com modelos utilizando regularização *à posteriori* (Ganchev et al., 2010), aproveitando o facto de que a presença de palavras correspondentes a nomes de locais deve ser vista como um forte indicador para que o documento pertença a uma determinada classe. Ainda no que se refere a nomes de locais, importa notar que, muito embora a identificação de coordenadas geoespaciais para a totalidade de um documento possa fornecer uma forma conveniente de ligar textos a locais específicos, útil para diferentes aplicações, existem muitas outras aplicações que poderiam beneficiar da resolução completa das referências a locais individuais nos documentos (Leidner, 2007). As distribuições de probabilidade para as células, fornecidas pelo método de classificação, podem por exemplo ser usadas para definir uma confiança prévia na resolução de nomes de locais.

Outra possibilidade de trabalho futuro relaciona-se com o uso de um meta-algoritmo originalmente proposto para problemas de regressão ordinal (Pang e Lee, 2005), ou seja, para problemas onde temos uma ordem natural entre os possíveis resultados, tais como o nosso problema em que temos uma noção de distância entre as células possíveis. A ideia básica deste método é a de que dados semelhantes devem receber classificações semelhantes. Usando esta premissa, podemos corrigir os resultados fornecidos pelos classificadores, considerando as classes reais dos knn documentos de treino mais semelhantes, obtidos através da semelhança do cosseno sobre os vec-

tores de características, e utilizando as fórmulas de Vincenty para medir a similaridade entre as classes (isto é, a distância entre as coordenadas centroide associadas às células). O classificador poderia ser usado como uma função inicial de preferência $\pi(x, l)$ que desse uma estimativa sobre a forma de classificar os documentos (i.e., que desse pontuações de classificação para um documento x e uma classe l). Essencialmente, iríamos utilizar uma métrica de distância entre as etiquetas d, e um conjunto de documentos $knn(x)$ com os knn exemplos mais próximos do documento x, de acordo com uma função de similaridade $sim(x, y)$ entre pares de documentos x e y. O problema da classificação de documentos pode ser resolvido através da escolha das classes que minimizem a fórmula abaixo, onde α representa um parâmetro de combinação ajustado empiricamente:

$$\sum_{x \in teste} \left[-\pi(x, l) + \alpha \sum_{y \in knn(x)} d(l_x, l_y) sim(x, y) \right] \quad (2)$$

Finalmente, gostaríamos também de experimentar com técnicas de expansão de documentos, especialmente para documentos pequenos, por forma a construir pseudo-documentos através da concatenação dos conteúdos relacionados com os documentos originais (por exemplo, utilizando as hiperligações entre documentos). Importa no entanto referir que estamos principalmente interessados na geocodificação de documentos usando apenas o texto, pois existe um grande número de situações (e.g., documentos históricos em bibliotecas digitais) em que outros tipos de informação simplesmente não estão disponíveis.

Agradecimentos

Este trabalho foi parcialmente financiado pela Fundação para a Ciência e Tecnologia (FCT), através dos projetos com referências PTDC/EIA-EIA/109840/2009 (SInteliGIS), UTA-Est/MAI/0006/2009 (REACTION), e PTDC/EIA-EIA/115346/2009 (SMARTIES). O autor Ivo Anastácio foi ainda suportado por uma bolsa de doutoramento com referência SFRH/BD/71163/2010.

Gostaríamos também de agradecer a Pável Calado, Luísa Coheur e Mário J. Silva, pelos seus comentários a versões preliminares deste trabalho.

Referências

Adams, B. e K. Janowicz. 2012. On the geo-indicativeness of non-georeferenced text. Em *Proceedings of the 6th International AAAI Conference on Weblogs and Social Media*.

Amitay, E., N. Har'El, R. Sivan, e A. Soffer. 2004. Web-a-where: geotagging web content. Em *Proceedings of the 27th Annual international ACM SIGIR Conference on Research and Development in Information Retrieval*.

Anastácio, I., B. Martins, e P. Calado. 2010. A comparison of different approaches for assigning geographic scopes to documents. Em *Proceedings of the 1st Simpósio de Informática*.

Bär, Hans e Lorenz Hurni. 2011. Improved density estimation for the visualisation of literary spacesl. *The Cartographic Journal*, 48.

Carpenter, Bob e Breck Baldwin. 2011. *Natural Language Processing with LingPipe 4*. LingPipe Publishing, draft edition.

Druck, Gregory, Gideon Mann, e Andrew McCallum. 2008. Learning from labeled features using generalized expectation criteria. Em *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in Information Retrieval*.

Dutton, G. 1996. Encoding and handling geospatial data with hierarchical triangular meshes. Em M. J. Kraak e M. Molenaar, editores, *Advances in GIS Research II*.

Eisenstein, Jacob, Brendan O'Connor, Noah A. Smith, e Eric P. Xing. 2010. A latent variable model for geographic lexical variation. Em *Proceedings of the Conference on Empirical Methods on Natural Language Processing*.

Erdmann, Eva. 2011. Topographical fiction: A world map of international crime fiction. *The Cartographic Journal*, 48.

Ganchev, Kuzman, João Graça, Jennifer Gillenwater, e Ben Taskar. 2010. Posterior regularization for structured latent variable models. *Journal of Machine Learning Research*, 11.

Gebel, Martin e Claus Weihs. 2007. Calibrating classifier scores into probabilities. Em *Proceedings of Advances in Data Analysis*.

Johnstone, B. 2010. Language and place. Em R. Mesthrie e W. Wolfram, editores, *Cambridge Handbook of Sociolinguistics*.

Kumar, Abhimanu, Matthew Lease, e Jason Baldridge. 2011. Supervised language modeling for temporal resolution of texts. Em *Proceeding of the 20th ACM conference on Information and Knowledge Management*.

Leidner, J. 2007. *Toponym Resolution in Text*. Tese de doutoramento, University of Edinburgh.

Lieberman, Michael D. e Hanan Samet. 2011. Multifaceted toponym recognition for streaming news. Em *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*.

Martins, B., I. Anastácio, e P. Calado. 2010. A machine learning approach for resolving place references in text. Em *Proceedings of the 13th AGILE International Conference on Geographic Information Science*.

Mehler, Andrew, Yunfan Bao, Xin Li, Yue Wang, e Steven Skiena. 2006. Spatial analysis of news sources. *IEEE Transactions Visualization in Computer Graphics*, 12.

Overell, Simon. 2009. *Geographic Information Retrieval: Classification, Disambiguation and Modelling*. Tese de doutoramento, Imperial College London.

Pang, Bo e Lillian Lee. 2005. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. Em *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. ACL.

Serdyukov, Pavel, Vanessa Murdock, e Roelof van Zwol. 2009. Placing flickr photos on a map. Em *Proceedings of the 32nd international ACM SIGIR conference on Research and development in Information Retrieval*.

Shakhnarovich, Gregory, Trevor Darrell, e Piotr Indyk. 2006. *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice (Neural Information Processing)*. The MIT Press.

Szalay, Alexander S., Jim Gray, George Fekete, Peter Z. Kunszt, Peter Kukol, e Ani Thakar. 2005. Indexing the sphere with the hierarchical triangular mesh. Relatório Técnico MSR-TR-2005-123, Microsoft.

Wing, Benjamin e Jason Baldridge. 2011. Simple supervised document geolocation with geodesic grids. Em *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*.

Análisis de la Simplificación de Expresiones Numéricas en Español mediante un Estudio Empírico

Susana Bautista
Universidad Complutense de Madrid. Facultad de Informática. Madrid, España
subautis@fdi.ucm.es

Biljana Drndarević
Universitat Pompeu Fabra. Department of Information and Communication Technologies. Barcelona, España
biljana.drndarevic@upf.edu

Raquel Hervás
Universidad Complutense de Madrid. Facultad de Informática. Madrid, España
raquelhb@fdi.ucm.es

Horacio Saggion
Universitat Pompeu Fabra. Department of Information and Communication Technologies. Barcelona, España
horacio.saggion@upf.edu

Pablo Gervás
Universidad Complutense de Madrid. Facultad de Informática. Madrid, España
pgervas@sip.ucm.es

Resumen

En este artículo se presentan los resultados de un estudio empírico llevado a cabo con un corpus paralelo de textos originales y simplificados a mano, y una posterior encuesta online, con el objetivo de identificar operaciones de simplificación de expresiones numéricas en español. Consideramos una "expresión numérica" como una frase que expresa una cantidad que puede venir acompañada de un modificador numérico, como por ejemplo *casi un cuarto*. Los resultados se analizan considerando las expresiones numéricas en oraciones con y sin contexto, a partir del análisis del corpus y del análisis de los resultados recogidos en la encuesta. Consideramos como trabajo futuro llevar a cabo una implementación computacional de las reglas de transformación extraídas.

Palabras clave

Simplificación de textos, Expresiones numéricas, Estudio de corpus

Abstract

In this paper we present the results of an empirical study carried out on a parallel corpus of original and manually simplified texts in Spanish and a subsequent survey, with the aim of targeting simplification operations concerning numerical expressions. For the purpose of the study, a "numerical expression" is understood as any phrase expressing quantity possibly modified with a numerical hedge, such as *almost a quarter*. Data is analyzed both in context and in isolation, and attention is paid to the difference the target reader makes to simplification. Our future work aims at computational implementation of the transformation rules extracted so far.

Keywords

Text Simplification, Numerical Expressions, Corpus Study

1 Introducción

Debido al crecimiento de Internet, cada vez más pronunciado, existe una tendencia para digitalizar todo tipo de información con el objetivo de hacerla más accesible a los usuarios. Sin embargo, los estudios demuestran que todavía estamos lejos de ese ideal de una sociedad digitalizada uniformemente donde la información sea asequible para todos. Ciertos usuarios, como las personas con trastornos visuales o auditivos, personas con bajo nivel de alfabetización, etc., se enfrentan con dificultades a la hora de acceder al contenido digital tal y como está presentado actualmente. Por ese motivo, ha habido mucho interés últimamente, por parte de distintas instituciones internacionales, para mejorar el esta-

do de accesibilidad de contenidos que se ofrecen en la Web con el fin de incluir a grupos actualmente marginalizados. La Organización de las Naciones Unidas (ONU) postula que todo el contenido que se publica en Internet debería ser accesible para las personas con discapacidad y hace referencia a Las Pautas de Accesibilidad de Contenido Web (Web Content Accessibility Guidelines, WCAG[1]), publicadas por un grupo de trabajo de W3C (World Wide Web Consortium). Sin embargo, según un estudio llevado a cabo por la ONU[2] con el objetivo de poner a prueba el estado de accesibilidad de un conjunto de 100 páginas web del mundo, sólo tres de ellas consiguen la accesibilidad básica prescrita por WCAG.

Muchos de los contenidos en la Web se presentan en forma escrita. Por lo tanto, la estructura y nivel de complejidad del texto escrito es un factor que influye en la accesibilidad de este tipo de contenidos. Muy a menudo, textos en la Web resultan demasiado complejos e incomprensibles para ciertos grupos de lectores, entre ellos personas con discapacidades cognitivas, personas con problemas de lectura o hablantes no nativos. Ha habido varios intentos de mejorar adecuadamente el contenido de lectura, bien a través de simplificaciones de materiales ya existentes o bien escribiendo material para un grupo objetivo específico. Ése es el caso, por ejemplo, de la Simple Wikipedia en inglés (Simple English Wikipedia[3]) y la Enciclopedia Elemental Británica (Encyclopedia Britanica Kids[4]) o el portal web en español de Noticias Fácil[5]. En España, existen distintas asociaciones y programas que apoyan la promoción de la Lectura Fácil, como la Asociación Lectura Fácil [6] en Barcelona y el programa de "Vive la fácil lectura"[7] en Extremadura. La lectura fácil contempla la adaptación a un lenguaje llano de textos legales y documentos informativos para instituciones y empresas que quieran mejorar la comunicación con su público destinatario, y promueve la edición de libros para personas con dificultades lectoras. En las simplificaciones, se considera el contenido, el lenguaje, las ilustraciones, y el diseño gráfico.

Sin embargo, la simplificación manual es demasiado lenta y costosa para ser una forma efectiva de producir la suficiente cantidad de material de lectura deseado. Por esta razón ha habido numerosos intentos de desarrollar sistemas de simplificación de textos automáticos o semi-automáticos, principalmente aplicados al inglés (Medero y Ostendorf, 2011), pero también japonés (Inui et al., 2003), portugués (Specia, 2010) y ahora español (Saggion et al., 2011). Estos sistemas utilizan técnicas computacionales en conjunto con los recursos lingüísticos para tratar tanto la estuctura sintáctica como el vocabulario del texto original que se ha de simplificar.

Nuestro trabajo sigue esta línea de investigación y se centra en esta contribución en las estrategias de simplificación léxica en textos informativos de género periodístico en español, con el objetivo de hacerlos más accesibles a las personas con discapacidad cognitiva. La importancia de las operaciones léxicas en la simplificación de textos ha sido ya tratada en trabajos previos (Carroll et al., 1998), (De Belder, Deschacht, y Moens, 2010), (Specia, 2010). El análsis del corpus que hemos llevado a cabo para el próposito de este artículo muestra también que los cambios léxicos son el tipo más común de todas las operaciones que aplican los editores humanos a la hora de simplificar un texto. En términos generales, las palabras y expresiones que se perciben como complicadas se cambian por sus sinónimos más simples o se parafrasean, como en el ejemplo que sigue (1 es la frase original, y 2 su simplificación)[8]:

1. *El Consejo de Ministros* **ha concedido** *hoy la Orden de las Artes y las Letras de España al* **restaurador** *José Andrés, a la escritora* **estadounidense** *Barbara Probst Solomon y al* **psiquiatra** *Luis Rojas Marcos.*

2. *Hoy el Gobierno de España* **ha dado** *el premio de la Orden de las Artes de España a tres personas. Al* **cocinero** *José Andrés, a la escritora* **de Estados Unidos** *Barbara Probst Solomon y al* **médico** *Luis Rojas Marcos.*

El primer cambio significativo es que la frase original ha sido divida en dos frases simplificadas. Además, en negrita se muestran los cambios observados en cuatro unidades léxicas.

En este trabajo nos centramos en un tipo particular de expresiones léxicas - las que con-

[1] http://www.w3.org/TR/WCAG/ [Último acceso: 20/11/2012]

[2] http://www.un.org/esa/socdev/enable/gawanomensa.htm [Último acceso: 20/11/2012]

[3] http://simple.wikipedia.org/wiki/Main_Page [Último acceso: 20/11/2012]

[4] http://kids.britannica.com/ [Último acceso: 20/11/2012]

[5] http://www.noticiasfacil.es/ES/Paginas/index.aspx [Último acceso: 20/11/2012]

[6] http://www.lecturafacil.net/content-management-es/ [Último acceso: 20/11/2012]

[7] http://www.facillectura.es/ [Último acceso: 20/11/2012]

[8] El ejemplo está extraído del corpus que describimos en la Sección 3.1

tienen información numérica. Consideramos una "expresión numérica" (ExpNum) como una frase que expresa una cantidad, opcionalmente acompañada de un modificador numérico, como son las expresiones: *más de un cuarto* o *cerca del 97 %*, donde *más de* y *cerca de* son ejemplos de modificadores numéricos. Este tipo de expresiones aparecen con una elevada frecuencia en el tipo de textos periodísticos que tratamos. A menudo las noticias diarias contienen información en forma numérica, y el modo en el que se presenta esta información afecta a la legibilidad de dichos textos. Consideremos la siguiente noticia, parte del corpus Simplext (ver Sección 3.1), y fijémonos en el número y la variedad de expresiones numéricas que contiene (marcadas en negrita):

> **CASI 400.000 PERSONAS** *DESPLAZADAS EN PAKISTÁN HAN VUELTO A CASA TRAS LAS INUNDACIONES*
>
> **Alrededor de 390.000 personas** *han regresado a sus casas desde que se vieran obligadas a desplazarse por las inundaciones causadas por las lluvias monzónicas del pasado verano en Pakistán. Según la Oficina de la ONU para la Coordinación de Asuntos Humanitario, esta cifra supone* **un 26 %** *de los* **1,5 millones de pakistaníes** *desplazados por las inundaciones. Por otro lado, la ONU ha logrado recaudar* **un 34 %** *de los* **2.000 millones de dólares** *(***cerca de 1.400 millones de euros***) solicitados como llamamiento de urgencia ante la catástrofe de Pakistán, la mayor petición realizada nunca por Naciones Unidas ante un desastre natural. Esta catástrofe ha matado a* **unas 2.000 personas**, *ha afectado a* **más de 20 millones**, *ha destruido* **cerca de 1,9 millones de hogares** *y ha devastado* **al menos 160.000 kilómetros cuadrados**, **una quinta parte del pas**. *Ante esta tesitura, el secretario general de la ONU, Ban Ki-moon, ha urgido a la comunidad internacional a responder "con generosidad y rapidez" a las necesidades humanitarias de Pakistán.*

En un texto relativamente corto encontramos hasta 12 expresiones numéricas distintas, que suponen dos expresiones numéricas por frase, en términos medios. Tanta carga informativa, al igual que la variedad de expresiones numéricas diferentes, pueden interferir con la comprensión del texto e impedirle al lector descubrir las relaciones de causa y efecto de los acontecimientos tratados en la noticia.

Por eso decidimos centrarnos en el tratamiento de las expresiones numéricas para la simplificación de textos en español. Este es un tema que no ha sido tratado en la literatura hasta ahora. Empezamos con un análisis de corpus, en el que observamos los cambios relativos a expresiones numéricas, hechos por humanos. De dicho corpus extrajimos un conjunto de expresiones numéricas y las presentamos en una encuesta, para que un grupo de participantes las simplificaran fuera de su contexto original. Nuestro objetivo es obtener un conjunto de operaciones para la simplificación de expresiones numéricas y plantear su implementación computacional, que sería una de las tareas en el proceso de simplificación de textos.

Este artículo está organizado como sigue: la Sección 2 presenta los trabajos relacionados en este área; en la Sección 3 describimos el conjunto experimental del estudio; el análisis de los datos es descrito en la Sección 4; la Sección 5 recoge nuestra discusión y conclusiones. Las líneas de trabajo futuro son presentadas en la Sección 6.

2 Trabajo Previo

Hasta ahora la simplificación de textos ha sido enfocada con dos objetivos diferentes. Uno es ofrecer versiones simplificadas de textos originales a grupos específicos de lectores humanos, como:

- estudiantes de lenguas extranjeras (Medero y Ostendorf, 2011);
- personas afásicas (Carroll et al., 1998), (Devlin y Unthank, 2006);
- personas con discapacidad auditiva (Inui et al., 2003);
- personas con bajo nivel de alfabetización (Specia, 2010), (Candido et al., 2009);
- personas no familiarizadas con textos técnicos altamente idiosincráticos tales como las patentes y los reglamentos (Bouayad-Agha et al., 2009).

Por otro lado, la simplificación de textos podría mejorar la eficiencia de otras tareas del procesamiento del lenguaje natural, tal y como se ha visto en los sistemas de traducción automática o en los sistemas de extracción de información (Chandrasekar, Doran, y Srinivas, 1996), (Klebanov, Knight, y Marcu, 2004).

De cualquier manera, la simplificación de texto hasta ahora ha afectado principalmente a las

construcciones sintácticas y a las expresiones léxicas percibidas como complejas o complicadas, como son oraciones largas con múltiples oraciones coordinadas y subordinadas, oraciones en voz pasiva, uso de palabras de baja frecuencia, palabras abstractas, términos técnicos y abreviaturas. Chandrasekar, Doran, y Srinivas (1996) y Siddharthan (2002) se centran principalmente en estructuras sintácticas, mientras que Carroll et al. (1998), dentro de su proyecto PSET (Practical Simplificacion of English Text) orientado hacia lectores con afasia, introducen también un módulo de simplificación léxica. Su enfoque se basa en búsqueda de sinónimos en WordNet en combinación con las frecuencias Kucera-Francis, extraídas de la base de datos Oxford Psycholinguistic Database (Quinlan, 1992). Por lo tanto, el sinónimo con mayor frecuencia dentro del conjunto de sinónimos extraídos para cada palabra léxica del texto original se escoge como su equivalente más simple.

Dicho enfoque basado en sinonimia y frecuencia de palabra ha sido reutilizado en varios trabajos. Lal y Ruger (2002) utilizan el mismo método para el componente léxico de su sistema de resumen automático. Burstein et al. (2007) se centran en los cambios de vocabulario a la hora de ofrecer su sistema ATA V.1.0 como herramienta para la adaptación de textos, pensada para los profesores y estudiantes de lenguas extranjeras. Su sistema produce párrafos resumidos del texto original, llamados notas marginales, y al mismo tiempo le ofrece al usuario sinónimos más frecuentes de palabras poco usadas, extraídos de WordNet calculando la similitud de palabras. Bautista, Gervás, y Madrid (2009) también emplean diccionarios de sinónimos, pero su criterio para escoger el más adecuado es longitud de palabra, en vez de la frecuencia.

Dado que muchas palabras, en particular las palabras con mayor frecuencia, tienden a ser polisémicas, se han visto varios intentos de tratar este problema con el objetivo de conseguir una sustitución léxica más precisa que también tenga en cuenta el contexto. Con este fin, De Belder, Deschacht, y Moens (2010) fueron los primeros en utilizar técnicas de desambiguación del sentido de las palabras. Para cada palabra léxica se crean dos conjuntos de "palabras alternativas" uno basado en sinónimos de WordNet o algún diccionario parecido, y otro generado con el modelo de lenguaje del análisis semántico latente (Deschacht y Moens, 2009). Una vez determinada la intersección de estos dos conjuntos, se calcula la probabilidad para cada palabra de la intersección con el fin de comprobar si dicha palabra es un reemplazo adecuado para la palabra de entrada. La probabilidad se calcula teniendo en cuenta la dificultad de la palabra basada en la frecuencia Kucera-Francis, el número promedio de sílabas y la probabilidad de cada palabra extraída de un corpus de textos de fácil lectura, tal como la Simple English Wikipedia.

Biran, Brody, y Elhadad (2011) emplean un método no supervisado de aprendizaje automático para aprender pares de sinónimos de palabras complejas y simples, basado en un corpus no alineado de textos de la Wikipedia original y la Wikipedia simple en inglés. Yatskar et al. (2010) también utilizan un método no supervisado para extraer simplificación léxica, utilizando el historial de ediciones de la Wikipedia simple en inglés.

En cuanto a las expresiones numéricas, existen algunos trabajos, aunque dirigidos principalmente a los expertos y no a los individuos con dificultades numéricas (Peters et al., 2007), (Dieckmann, Slovic, y Peters, 2009), (Mishra H, 2011).

Bautista et al. (2011) y Power y Williams (2012) se encuentran entre los primeros en concentrarse en la posibilidad de simplificar este tipo de expresiones, centrándose principalmente en el uso de modificadores. Power y Williams (2012) realizaron un estudio de un corpus de noticias en inglés, analizaron como los autores variaban las formas matemáticas y la precisión de las mismas cuando ellos expresaban información numérica. En un documento una misma cantidad era a menudo descrita de distintas maneras, variando su expresión (fracción, porcentaje) y su precisión, usando modificadores y redondeo para ello. Además, desarrollaron un sistema basado en restricciones para decidir como adaptar la proporción original. El trabajo de Bautista et al. (2011) estudia la preferencia de valores comunes a la hora de redondear las expresiones numéricas y el uso de diferentes estrategias de simplificación dependiendo del valor de la proporción original. Está desarrollado para textos en inglés, no fue dirigido a un grupo determinado de lectores, y la simplificación se realizó de acuerdo a los niveles de dificultad según se describen en el Currículo de Matemáticas de la Autoridad de Calificaciones y Currículum de Inglaterra (Qualifications y Authority, 2010).

3 Metodología y Objetivos

Con el objetivo de esbozar conclusiones sobre el tipo de operaciones de simplificación que podrían ser aplicadas a las expresiones numéricas, hemos llevado a cabo un estudio de un corpus paralelo de textos originales en español y su

correspondiente versión simplificada a mano. El estudio del corpus forma parte de un trabajo más amplio, cuyo objetivo es desarrollar un sistema para la simplificación autómatica de noticias en español. Desarrollando el módulo de la simplificación léxica, hemos observado un número elevado de expresiones numéricas y sus simplificaciones en el corpus. En un intento de investigar más a fondo el caso de la simplificación de dichas expresiones, las tratamos como un caso específico de la simplificación léxica y las analizamos por separado.

Con el fin de ampliar el conjunto de las posibles simplificaciones relacionadas a estas expresiones, llevamos a cabo una encuesta complementaria al estudio del corpus. Las expresiones numéricas del corpus han sido etiquetadas y extraídas, junto con el resto de la frase donce aparecen, para presentarlas de manera separada en dicha encuesta. A los participantes de la encuesta se les pidió que simplificaran las expresiones numéricas que se les ofrecieron.

Por lo tanto, por un lado tenemos expresiones numéricas en contexto, es decir, en el corpus, donde se pueden observar otras operaciones de simplificación, como por ejemplo sustituciones basadas en sinonimia o reestructuración sintáctica. Además de eso, el corpus fue simplificado por expertos teniendo en mente como usuario final un lector específico - una persona con dificultades lectoras debido a discapacidades cognitivas. Por otro lado, se extrajeron oraciones individuales del mismo corpus que contienen expresiones numéricas, y se presentaron fuera de contexto a los participantes de la encuesta para que las simplificaran, sin tener en cuenta quién era el usuario final. El objetivo es ampliar el conjunto de posibles operaciones de simplificación de las expresiones numéricas, no necesariamente relacionadas a un género de texto o a un usuario final dado. En el caso de la encuesta, estas simplificaciones fueron libres, en el sentido que fueron simplificadas sin especificar ningún grupo objetivo de lectores, por lo que los participantes simplificaron de manera general.

Dentro de la variedad de tipos encontrados en las expresiones numéricas, hemos limitado nuestro trabajo al tratamiento de expresiones monetarias (*15 millones de euros*), porcentajes (*24 %*), fracciones (*un cuarto*), dimensiones físicas (*160,000 kilómetros cuadrados*) y cantidades generales (*2,000 personas*). En la sección 4.3 se discute cómo las simplificaciones hechas en el corpus y en la encuesta difieren y se complementan unas a otras, con la intención de obtener conclusiones para la posible implementación computacional de la simplificación de expresiones numéricas. A continuación describimos el conjunto de datos experimental, al igual que los recursos empleados para el análisis - el corpus, las herramientas del procesamiento del texto y la encuesta.

3.1 Corpus

Como parte de un proyecto más amplio[9], orientado hacia el desarrollo de un sistema de la simplificación automática de textos en español para los lectores con discapacidad cognitiva, hemos recopilado un corpus paralelo para usar como base para un análisis empírico. Dicho corpus consiste en 40 textos informativos, en el dominio de noticias internacionales y de cultura, cedidos por la agencia española de noticias Servimedia[10]. Los textos han sido simplificados por editores humanos, teniendo en cuenta el usuario final - un lector con discapacidad cognitiva, y siguiendo una serie de pautas de la metología de fácil lectura sugerida por Anula (2007), (2008). Dichas pautas incluyen una serie de reglas, que se podrían resumir de la siguiente manera:

- tratamiento de la microestructura del texto, es decir la estructura de la frase y los elementos del vocabulario;
- tratamiento de la información, como la reducción o expansión del contenido;
- tratamiento del discurso, como el estilo;
- la aplicación de una adecuada norma ortográfica.

Ambos conjuntos de textos, original y simplificado, han sido anotados automáticamente usando las etiquetas del procesamiento morfológico de las palabras, el reconocimiento de entidades nombradas y el análisis sintáctico, proporcionados por el paquete de análisis de lenguaje de FreeLing (Padró et al., 2010), descrito con más detalle en la sección 3.2. Además de esto, un algoritmo de alineación de textos (Bott y Saggion, 2011) ha sido aplicado para conseguir alineación a nivel de oración entre los textos originales y simplificados. Los errores de alineación han sido manualmente corregidos usando una herramienta gráfica de edición en el marco de GATE (General Architecture for Text Engineering) (Maynard et al., 2002).

[9] www.simplext.es [Último acceso: 20/11/2012]

[10] http://www.servimedia.es/ [Último acceso: 20/11/2012]

De esta manera hemos obtenido un corpus paralelo de un total de 570 oraciones, 246 en el conjunto original y 324 en el conjunto simplificado. Dicho corpus nos ha servido para documentar todas las operaciones de edición aplicadas por los humanos para planificar y organizar su implementación automática. Entre la variedad de operaciones detectadas actualmente nos centramos en simplificaciones léxicas, más específicamente en el tratamiento de las expresiones numéricas, que es el trabajo que presentamos en este artículo.

3.2 Procesamiento del texto

Tal y como mencionamos en el párrafo anterior, los textos del corpus han sido analizados usando FreeLing (Padró et al., 2010) y después procesados con la herramienta de edición de textos GATE (General Architecture for Text Engineering) (Maynard et al., 2002). GATE es un conjunto de herramientas para el procesamiento de lenguaje natural que se integran en una plataforma escrita en Java. Dispone de una interfaz gráfica y un entorno de desarrollo integrado que facilita considerablemente las tareas que requieren un proceso de edición y editores especializados. GATE es de distribución libre y de código abierto.

FreeLing es una de las herramientas de análisis del procesamiento de lenguaje natural existentes para el castellano que permite realizar análisis morfológico (part-of-speech tagging) basado en un modelo de Markov con estados ocultos. Este tipo de análisis anota los textos e identifica los lemas de cada palabra, asignándole su correspondiente etiqueta. El sistema de etiquetado usado por FreeLing sigue el estándar EAGLES[11]. Para el propósito de este artículo nos hemos centrado en las etiquetas correspondientes a expresiones numéricas. A las cifras y a los números se les asigna la etiqueta Z. Bajo esta etiqueta podemos encontrar números, ratios, porcentajes, dimensiones, etc. FreeLing identifica cuatro tipos distintos de numerales que etiqueta de manera distinta:

1. Los numerales partitivos tienen la etiqueta Zd (p.e. *una docena*, *un millón*, *un centenar*, etc.).

2. Las cantidades monetarias reciben la etiqueta Zm, que tienen como lema la cantidad (en cifras) y el nombre de la unidad monetaria en sigular (p.e. *2000 dólares*, cuyo lema es $_USD:2000)

3. Las fracciones y porcentajes tienen la etiqueta Zp. El lema normaliza la proporción (p.e. *74 %*, cuyo lema es 74/100)

4. Las magnitudes físicas reciben la etiqueta Zu. El lema normaliza la unidad de medida y la magnitud (p.e. *30Km/h*, cuyo lema es SP_km/h:30).

Para empezar, usamos FreeLing para el análisis morfológico del corpus, y una vez que los textos están etiquetados, llevamos a cabo la tarea de anotación de las expresiones numéricas en GATE. Para hacer posible la integración de ambas herramientas, es necesario convertir el formato de salida de FreeLing en un formato XML legible por GATE.

Para anotar las diferentes expresiones numéricas en los textos originales, incluyendo sus posibles modificadores, hemos utilizado GATE para definir un conjunto de gramáticas JAPE (Java Annotation Patterns Engine). JAPE es una versión de CPSL - Common Pattern Specification Language. JAPE proporciona la traducción de estados finitos sobre anotaciones basadas en expresiones regulares y reconoce las expresiones regulares en las anotaciones en los textos que queremos analizar. Una gramática JAPE contiene conjuntos de reglas, organizadas en fases y compuestas por patrones y sus correspondientes acciones. Las fases se ejecutan en cascadas de transductores de estados finitos sobre las anotaciones en los textos originales. La parte izquierda de la regla (left-hand-side, LHS) describe el patrón de la anotación, mientras la parte derecha de la regla (right-hand-side, RHS) sirve para declarar qué acciones ejecutar sobre la anotación en cuestión. Es posible hacer referencia a las anotaciones de LHS en la parte de la derecha, poniéndoles etiquetas a los elementos del patrón.

En la Figura 1 se puede ver un ejemplo de un texto original del corpus con las expresiones reconocidas usando las gramáticas JAPE definidas para anotar los distintos tipos de expresiones numéricas. El Cuadro 1 muestra un ejemplo de la regla titulada “CasiPorcFract”, que usamos para identificar las expresiones numéricas de tipo porcentajes y fracciones acompañadas por el modificador “casi”. La parte que precede a “–>” es la parte izquierda, y la parte derecha es la parte que le sigue. La parte izquierda especifica un patrón que tiene que coincidir con las anotaciones que existen en el documento GATE, mientras que la parte derecha especifica que es lo que hay que hacer con el texto coincidente. En el ejem-

[11] http://nlp.lsi.upc.edu/freeling/doc/tagsets/tagset-es [Último acceso: 20/11/2012]

Figura 1: Ejemplo de texto con las expresiones reconocidas usando gramáticas JAPE

plo, la regla tiene el título "CasiPorcFract", la cual comprueba en el texto anotado las palabras que tienen en su lema una característica "casi" y la palabra está anotada con la etiqueta "Zp". Una vez que la regla ha encontrado una secuencia de texto que coincida con este patrón, la anota con la etiqueta que se indica después de la palabra "annotate" en la parte derecha de la regla, en este caso, con la etiqueta "CASIporcFract". Además, dentro de la expresión numérica identificada, se etiqueta como MOD_EXP el texto que corresponde con el modificador y que ha sido identificado en la parte izquierda con la etiqueta "modifier". De esta forma, tendremos anotado dentro de la expresión numérica tanto el modificador como la cantidad. El texto queda anotado con la gramática JAPE definida para este tipo de expresión "CasiPorcFract", cuyo modificador es "casi", acompañado de cualquier cantidad etiquetada por "Zp", como se puede ver en el ejemplo de la Figura 2, para el caso de "Casi el 20 %".

Rule: CasiPorcFract
(((word.lemma="casi") (word)?): **modifier**
(word.tag="Zp")):**annotate**
–>
:**modifier**.MOD_EXP={semantics="casi"},
:**annotate**.CASIporcFract= {semantics="porcFract"}

Cuadro 1: Ejemplo de una regla de una gramática JAPE

Estas gramáticas en GATE las usamos para

Figura 2: Ejemplo de expresión numérica anotada correspondiente a la regla JAPE mostrada

anotar todos los distintos tipos de expresiones numéricas que encontramos en el corpus. Esto nos permite llevar a cabo un análisis del corpus e identificar diferentes tipos de expresiones para ser presentadas en la encuesta a los participantes. Para desarrollar las reglas hemos contado con el sistema ANNIC (Aswani et al., 2005), y un componente de GATE para indexación, anotación y búsqueda. Este sistema nos permite hacer búsqueda en el corpus anotado con las etiquetas de nuestro interés, que han sido generadas a partir de las reglas que hemos definido en nuestras gramáticas. Este conjunto de gramáticas JAPE es un primer paso para una futura implementación de las reglas de simplificación.

Sobre un subconjunto de 10 textos, con un total de 59 oraciones, pertenecientes al corpus se lleva a cabo la correción manual de las reglas ejecutadas automáticamente. Usando la herramienta GATE se hace una com-

paración automática identificando las etiquetas nuevas creadas manualmente y las generadas automáticamente a partir de las gramáticas JAPE definidas. Las gramáticas desarrolladas utilizando el método previamente explicado tienen una cobertura de 13 casos diferentes de expresiones numéricas de los cuatro tipos distintos identificados por el analizador. En el Cuadro 2 mostramos los 13 casos identificados en el corpus usado para medir la cobertura de las reglas definidas.

Hemos comprobado el rendimiento de las reglas definidas y hemos obtenido los siguientes resultados globales: precision= 0.94, recall= 0.93 y F-measure= 0.93. Para cada etiqueta, GATE calcula, precision, recall y F-measure, y hemos observado que en las expresiones numéricas menos frecuentes se obtienen peores resultados pero para las expresiones numéricas más frecuentes se obtienen muy buenos resultados. En los resultados globales vemos que tenemos una precision y un recall muy altos, ya que nuestras reglas etiquetan una fracción bastante alta de las instancias relevantes del corpus.

3.3 Encuesta

El objetivo de la encuesta es ampliar el conjunto de posibles operaciones de simplificación de expresiones numéricas obtenidas del corpus. Oraciones aisladas que contienen expresiones numéricas se les ofrecen a los participantes en la encuesta para que propongan sus propias simplificaciones.

Para ello, se preparó un cuestionario usando la herramienta que proporciona Google para hacer formularios, y se albergó en Google Docs[12]. La evaluación experimental incluyó a 23 participantes, todos hablantes nativos de español en posesión de un título universitario. El cuestionario se compone de frases tomadas de la recopilación antes mencionada, con la diferencia de que el contexto que las rodea fue omitido y el único cambio que se aplica es el relativo a las expresiones numéricas que se traten en cada oración. Para este cuestionario se optó por 14 frases con un total de 27 expresiones numéricas. Doce de las expresiones originales ya contenían un modificador, mientras que las 15 restantes no lo contenían. La siguiente frase es un ejemplo del tipo de oraciones que se presentaron en la encuesta:

> *Esta catástrofe ha matado a [***unas 2.000 personas***], ha afectado a [***más de 20 millones***], ha destruido [***cerca de 1,9 millones de hogares***] y ha devastado [***al menos 160.000 kilómetros cuadrados***], una [***quinta parte***] del país.*

Los participantes tenían que proporcionar simplificaciones de las expresiones numéricas marcadas por corchetes en cada frase que se presentaba en el cuestionario. Las instrucciones decían que las expresiones numéricas se podían simplificar utilizando cualquier formato: números en palabras, cifras, fracciones, proporciones, etc. Así mismo, se indicó que los modificadores tales como *menos que* o *alrededor de* podían ser utilizados si se consideraba necesario. A los participantes se les indicó que mantuvieran el sentido de la frase en la versión simplificada tan cerca como fuese posible del sentido de la oración original y que, de ser necesario, se podía reescribir la sentencia original completa. No se impusieron más restricciones, es decir, los usuarios no recibieron instrucciones para aplicar las reglas de simplificación que se habían extraído previamente del corpus, dado que la idea era compararlas con las operaciones extraídas del corpus y estudiar dicha comparación. La Figura 3 muestra una pequeña parte de la encuesta, donde se puede ver una oración que se presentó a los usuarios, con una expresión numérica entre corchetes, la cuál se pedía simplificar.

4 Análisis de los datos

Aquí presentamos los resultados obtenidos por separado: en primer lugar, a partir del análisis del corpus, y en segundo lugar, a partir del análisis de los resultados recogidos en la encuesta realizada. Los datos obtenidos se analizan con un enfoque comparativo, con el objetivo de extraer conclusiones sobre la posibilidad de la implementación de las reglas de simplificación extraídas.

4.1 Análisis del corpus

Como ya se ha mencionado, aquí tratamos expresiones numéricas como casos específicos de simplificación léxica. El análisis del corpus, compuesto por textos periodísticos, que se llevó a cabo con el fin de extraer las estrategias de simplificación léxica, ha mostrado que las expresiones numéricas no sólo son abundantes en este género, sino que también se modifican con frecuencia para conseguir un texto de salida más fácil de leer. Cada texto original contiene un promedio de 3,78 expresiones numéricas.

[12] https://docs.google.com/spreadsheet/viewform?formkey=dDhWQ2NyckpUTUthbTVIRVVFTUtaRGc6MQ#gid=0 [Último acceso: 20/11/2012]

Etiqueta	Expresión Numérica	Ejemplo
CASIporcFract	casi + Zp	casi un cuarto
DURANTENUM	durante + Z	durante 24 das
MASDENUM	más de + Z	más de 50.000
MASDEPART	más de + Zd	más de 20 millones
MASDEporcFract	más de + Zp	más del 40 %
NUMERALES	Z	34.589
NUMMAGNITUDES	Zu	32 metros
NUMMONETARIAS	Zm	1.400 euros
NUMPARTITIVO	Zd	32 millones
NUMPORCENTYFRACT	Zp	75 %
UNASMagnit	unas + Zu	unas 700 millas
UNASNUM	unas + Z	unas 20.000
MOD_EXP	modifier	alrededor, menos de...

Cuadro 2: Tipos identificados en el corpus usado para medir la cobertura de las reglas

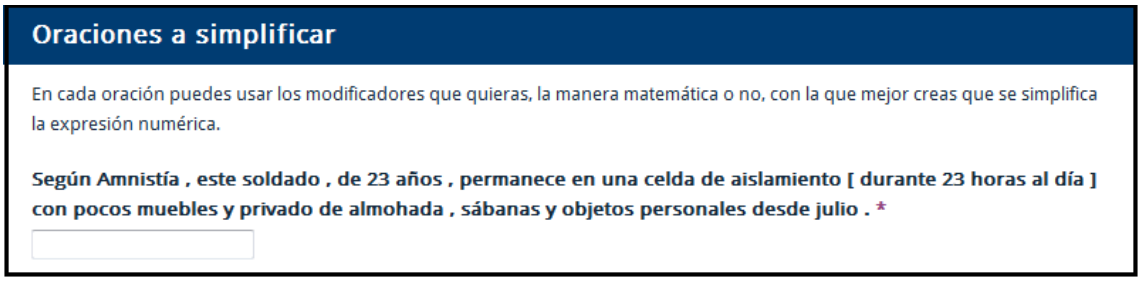


Figura 3: Ejemplo de un parte de la encuesta

En las versiones simplificadas de los textos, un número significativo de estas expresiones numéricas son eliminadas: haciendo el cálculo, menos de la mitad de estas expresiones en los textos originales se han conservado en sus versiones simplificadas. De las expresiones que no se eliminan, la mayoría contienen algún tipo de modificación y en el texto simplificado se presentan de forma diferente a la que aparece en el texto original. También hemos observado un uso variado de modificadores, entre ellos, *más de*, *cerca de*, *casi*, etc.

Ha habido casos en que las expresiones numéricas son eliminadas, en otros casos el número original se redondea cuando una expresión es sustituida por otra, o casos en que el número fue redondeado usando además un modificador añadido que no estaba presente en el texto original. En los trabajos previos (Bautista et al., 2011), (Power y Williams, 2012) ya se sugiere que los modificadores pueden ser una herramienta útil para simplificar una variedad de diferentes expresiones numéricas.

Lo que sigue es un resumen de las operaciones de simplificación más comunes aplicadas a expresiones numéricas en el corpus:

1. Los números en parentésis se eliminan (esta operación ha sido aplicada en un 100 % de los casos en la simplificación manual):

   un millón de francos suizos (unos 770.000 euros) ⇒ *un millón de francos suizos*

2. Los números en letras se sustituyen por números expresados con dígitos:

   nueve millones ⇒ *9 millones*

3. Las grandes cantidades se expresan por medio de una palabra en lugar de dígitos:

   unos 370.000 niños ⇒ *más de 300 mil niños*

4. Grandes números se redondean:

   casi 7.400 millones de euros ⇒ *más de 7000 millones de euros*

5. Se aplica redondeo eliminando puntos decimales:

   1,9 millones de hogares ⇒ *2 millones de casas*[13]

Tras el análisis del corpus, teniendo en mente una futura implementación computacional de las reglas identificadas, se lleva a cabo una encuesta dirigida exclusivamente a la simplificación de expresiones numéricas para observar el uso de modificadores y las estrategias de simplificación

[13] Aquí otro cambio léxico es aplicado: hogar ⇒ casa

aplicadas. Recopilando esta información, podemos completar los resultados obtenidos en el estudio de corpus antes mencionado de cara a la implementación.

4.2 Resultados de la encuesta

Los datos recogidos a partir de la encuesta realizada han sido analizados para identificar las operaciones de simplificación que los participantes han usado para simplificar las expresiones numéricas.

Para cada expresión numérica en una oración dada identificamos todas las operaciones usadas por todos los participantes. Se han identificado un total de 26 operaciones diferentes aplicadas para simplificar las expresiones dadas en la encuesta. Algunos ejemplos son añadir una explicación, calcular el tanto por ciento dado, cambiar de porcentaje a fracción, etc. No todas las operaciones ocurren con suficiente frecuencia como para tenerlas en cuenta en el análisis, por lo que han sido agrupadas dependiendo del tipo de cambio aplicado (por ejemplo si han usado o no modificador) o si la información ha sido eliminada, la cantidad redondeada o la expresión númerica reescrita. Por eso, nos centramos en las operaciones más comunes aplicadas por los participantes.

Como ilustración, veamos el ejemplo de la expresión original *55* en la frase:

> *Amnistía Internacional ha documentado durante 2010 casos de tortura y otros malos tratos en al menos 111 países, juicios injustos en **55**, restricciones a la libertad de expresión en 96 y presos de conciencia encarcelados en 48.*

Las siguientes simplificaciones fueron sugeridas por los sujetos:

- *más de 50*
- *más de la mitad de ellos*
- *la mitad de ellos*
- *55*
- *50*

La expresión simplificada más comúnmente usada fue *más de 50*, donde un modificador es añadido y el número redondeado, aunque con una pequeña pérdida de precisión.

Las observaciones generales que sacamos del análisis de datos obtenidos del cuestionario son las siguientes:

- El número en sí mismo:
  - se deja sin cambios *(26.3 %)*,
  - se redondea *(26.3 % ⇒ más de un 25 %)*,
  - se cambia su forma matemática *(24 % ⇒ casi un cuarto)*,
  - se reescribe en letras *(3 % ⇒ tres por ciento)*,
  - se reescribe en dígitos *(ocho millones ⇒ 8 millones)*
- En ocasiones se pierde precisión de la expresión numérica cuando se sustituye por una versión simplificada. Por ejemplo, *Alrededor de 390.000 personas ⇒ Casi 400.000 personas*
- Si la expresión original no tenía modificador, en ocasiones un modificador es usado en la opción simplificada para tener en cuenta la pérdida de precisión. Por ejemplo, *78 % ⇒ más del 75 %*

En las oraciones presentadas en la encuesta estudiamos, por un lado, las expresiones originales que ya contienen un modificador y, por otro, las que van sin modificador. De las 27 expresiones numéricas originales presentadas en la encuesta, 15 de ellas no tenían modificador mientras que las restantes 12 sí tenían.

En el caso de las 12 expresiones originales con modificadores, en 7 de ellas la operación de simplificación usada más común fue sustituir el modificador original por otro y redondear el número. Esto ocurre con los siguientes modificadores: *al menos* y *casi* son sustituidos por *más de*, mientras que *unos*, *alrededor de* y *cerca de* son sustituidos por *casi*. En 4 expresiones, el modificador original se mantuvo sin cambios, como es el caso de *más de*, *unos* o *unas*, mientras que el número fué redondeado. Hubo sólo un caso donde la expresión numérica original fue completamente reescrita por la mayoría de los participantes en la encuesta y por lo tanto el modificador original se perdió.

Por otro lado, de las 15 expresiones numéricas originales sin modificador, en 8 casos un modificador fue añadido por la mayoría de los participantes; 5 casos continuaron sin modificador (todos ellos debido al hecho de que la simplificación es igual a la original, es decir, no hubo ningún cambio); y en 2 casos la operación más común fue reescribir la expresión numérica original.

Consideramos como casos de reescritura los casos en los que se eliminó la expresión numérica original y se utilizó información textual en su lugar, tal como en el ejemplo siguiente: *durante 23*

horas al día se reescribió como *casi todo el día*. Además, observamos simplificaciones donde un cambio de estrategia de simplificación fue aplicado, como se pueden ver en estos ejemplos: la expresión *26 %* fue simplificada usando una expresión en forma de fracción *una cuarta parte*, y lo mismo fue aplicado en el caso de *34 %*, el cuál fue reescrito como *un tercio*. Los resultados de la encuesta nos hacen ver que el uso de modificadores juega un papel fundamental cuando se simplifican expresiones numéricas.

Nuestros datos muestran que las operaciones más comúnmente aplicadas son añadir un modificador cuando la expresión original no lo tiene ya, y redondear la expresión numérica original, explicado en profundidad en la Sección 4.3.

4.3 Análisis comparativo

Para llevar a cabo un análisis comparativo de los resultados obtenidos en el estudio realizado sobre el corpus y sobre la encuesta, nos centramos en el subconjunto de expresiones númericas usadas en la encuesta y en sus equivalentes en el corpus. Posteriormente, hemos extraído todas las operaciones aplicadas en el proceso de simplificación de las expresiones seleccionadas y comparamos las frecuencias relativas de estas operaciones en el corpus y en la encuesta. Los Cuadros 3 y 4 presentan los resultados. Las filas marcadas corresponden a las operaciones que coinciden en ambos casos.

Operaciones de simplificación	**Número de ExpNum**	**% Uso**
Eliminar ExpNum	12	44.4 %
Eliminar Oración	7	25.9 %
Misma ExpNum	2	7.4 %
Cambiar Modificador + Redondeo	2	7.4 %
Eliminar Modificador + Redondeo	2	7.4 %
Reescribir ExpNum	1	3.7 %
Eliminar Modificador + Mismo número	1	3.7 %
Total	**27**	100 %

Cuadro 3: Operaciones de simplificación obtenidas del análisis del corpus

En los resultados obtenidos del análisis del corpus, más del 50 % de las expresiones numéricas fueron eliminadas, mientras que los resultados de la encuesta sugieren una preferencia por mantener la información a costa de una ligera pérdida de precisión a través de redondeos y compensada por el uso de modificadores. En comparación con la simplificación del corpus, se opta más a menudo por reescribir la información o dejar las expresiones sin modificar, principalmente en los casos de los números grandes como *2.000 millones de dólares*, *más de 20 millones* o *65 millones*.

Operación de simplificación	**Número de ExpNum**	**% Uso**
Añadir Modificador + Redondeo	9	33.3 %
Cambiar Modificador + Redondeo	6	22.2 %
Misma ExpNum	5	18.5 %
Reescribir ExpNum	5	18.5 %
Mantener Modificador + Redondeo	2	7.4 %
Total	**27**	100 %

Cuadro 4: Operaciones de simplificación obtenidas del análisis de la encuesta

En cuanto al uso de los modificadores, los datos recogidos de la encuesta muestran que los modificadores preferidos cuando una expresión numérica se simplifica son: *más de* y *casi*. Estos dos modificadores han sido los más utilizados tanto cuando el modificador de la expresión original se cambia por otro, como cuando el modificador se añade a la expresión ya que inicialmente ésta no contenía ningún tipo de modificador.

Observando las operaciones de simplificación aplicadas por los participantes tanto en la simplificación del corpus como en la encuesta, se puede ver que hay tres operaciones comunes en ambos casos: *Cambiar Modificador + Redondeo, Misma ExpNum* y *Reescribir ExpNum*. La primera y la segunda tienen un uso similar. Obviando los casos de eliminación del corpus, son las dos operaciones más usadas por los expertos en la simplificación de las oraciones con contexto. Y en el caso de la encuesta, sin contar el caso más usado (*Añadir Modificador + Redondeo*), estas operaciones son también muy usadas por los participantes para simplificar las oraciones sin contexto. De ahí que, dependiendo del tipo de la expresión numérica original, una u otra sean usadas para proporcionar una expresión simplificada. En el caso de la última operación, *Reescribir ExpNum*, es mucho más frecuente en el caso de la simplificación de oraciones sin contexto en comparación con el caso de los textos del corpus.

Además, es significativo destacar que de las operaciones no comunes en los dos análisis, en el caso del corpus todas ellas están relacionadas con la eliminación de información (oraciones, expresiones numéricas, modificadores) y en cambio,

en el caso de la encuesta se añade información o se lleva a cabo una transformación de la expresión, manteniendo el modificador pero aplicando un redondeo a la cantidad. Uno de los factores que influye a la hora de detectar tantos casos de eliminación en el caso de la simplificación del corpus, es que cuando se pide simplificar un texto en seguida se asocia con la idea de eliminar información superflua para que así sea más fácil de leer y comprender. Pero esto no siempre es así, ya que la pérdida de información no garantiza un texto más simple. A veces hay que añadir información para ayudar a la lectura y comprensión del texto y entran en juego otros factores, como la frecuencia de uso de las palabras, la ambigüedad y el uso en el contexto de las mismas.

Durante el análisis de las simplificaciones sugeridas por los participantes de la encuesta, detectamos que para algunas de las opciones simplificadas que propusieron el contexto de la expresión numérica dentro de la oración había sido considerado. Veamos por ejemplo en la oración: *Amnistía Internacional ha documentado durante 2010 casos de tortura y otros malos tratos en al menos 111 países, juicios injustos en **55**, restricciones a la libertad de expresión en 96 y presos de conciencia encarcelados en 48.* Para la expresión original **55**, de los casos mostrados en la sección 4.2, podemos observar que dos de las simplificaciones (*más de la mitad de ellos, la mitad de ellos*) han sido propuestas simplificando la expresión original considerando el contexto a nivel de oración y haciendo referencia a los "111 países" nombrados anteriormente. Esto es significativo, porque a pesar de que las oraciones fueron presentadas sin contexto respecto al texto completo, algunas simplificaciones de expresiones numéricas propuestas por los participantes sí que consideraron el contexto a nivel de oración para generar una versión simplificada.

5 Discusión y Conclusiones

Los casos de eliminación, de la oración entera o justo de la expresión numérica en concreto, sólo aparecen en el análisis del corpus. Esto se debe al hecho de que los ejemplos dados en la encuesta eran oraciones individuales sin información añadida, mientras que los ejemplos en el corpus siempre van acompañados por contexto. Por lo tanto, en las oraciones de la encuesta no se producen casos de eliminación de la expresión numérica, y menos de la oración completa, ya que no se daba información añadida de donde aparecía la oración en el texto original.

Además hay que señalar que no se dió como posibilidad a los participantes la opción de eliminar información, solo de simplificar las expresiones que aparecían en cada oración. Estos casos ponen de relieve el papel importante que juega el contexto a la hora de decidir si eliminar o modificar una expresión numérica en una oración.

La simplificación manual del corpus se hizo sabiendo que el lector final sería una persona con discapacidad cognitiva mientras que en la encuesta no se especificó ningún usuario final a quien iban dirigidas las simplificaciones de las oraciones que se presentaban. Por lo tanto, lo que se tiene que decidir es si se debe dar preferencia a la preservación de la información a coste de la precisión, o eliminar la información superflua por completo de un texto que contiene expresiones numéricas.

El corpus que hemos utilizado en este trabajo, ha sido simplificado teniendo en cuenta el contexto y con conocimiento del usuario final a quien iba dirigida la simplificación. Estos dos factores permiten una eliminación selectiva con pérdida muy controlada de información (porque al usuario no le va a servir o porque ya se extrae del contexto).

Dentro del conjunto de operaciones de simplificación identificadas, observamos que hay operaciones comunes a la hora de simplificar las expresiones numéricas teniendo en cuenta el contexto (corpus) y sin tener en cuenta el contexto del texto (encuesta). Lo que demuestra que hay operaciones que, a priori, son más independientes del contexto, y que se aplican en ambos casos, obteniendo una versión simplificada de la expresión numérica que se quiere adaptar.

Es significativo que usando el analizador Freeling seamos capaces de identificar y anotar diferentes tipos y muchos casos distintos de expresiones numéricas, ya que en comparación con otros analizadores basados en aprendizaje automático como, OpenNLP[14], Maltparser[15], Mate-tools[16], que basan su análisis en el corpus que se utiliza para su entrenamiento, y usan la anotación del Penn Treebank POS, en la que sólo se dispone de una única etiqueta para categorías gramaticales (POS) para la información numérica que es *CD*, no pueden dar mayor detalle de qué tipo de expresión numérica ha sido identificada.

Este estudio realizado corrobora las conclusiones previas de los trabajos de Bautista et al.

[14] http://opennlp.apache.org/documentation.html [Último acceso: 20/11/2012]

[15] http://www.maltparser.org/ [Último acceso: 20/11/2012]

[16] http://code.google.com/p/mate-tools/ [Último acceso: 20/11/2012]

(2011) y Power y Williams (2012), sobre el uso de modificadores y el uso de distintas estrategias de simplificación, en este caso para la adaptación de textos en español.

6 Trabajo Futuro

Como parte de nuestro trabajo futuro tenemos la intención de reunir un corpus más rico en expresiones numéricas variadas y repetir el estudio con los editores humanos con el fin de extraer más posibles operaciones de simplificación para otros tipos de expresiones aquí no tratadas, como son por ejemplo el tratamiento de los porcentajes.

Además de esto, tenemos planeado incluir información sobre el usuario final para el que se está simplificando como un factor más a tener en cuenta, ya que las simplificaciones pueden variar dependiendo de para quién se simplifique el texto original. Si se opta por perder precisión, preservarla o eliminar la información que no sea necesaria, tomar estas decisiones en gran medida depende del tipo de lector para el que vaya destinado el texto simplificado.

Desde el punto de vista de eliminación de información, un posible enfoque es utilizar técnicas de resumen automático para desarrollar un clasificador que se pueda emplear como herramienta para la simplificación de textos, y ayude a decidir qué contenido guardar y qué elementos borrar, donde el número de expresiones numéricas se utiliza como un rasgo para crear el clasificador (Drndarević y Saggion, 2012).

El último objetivo de nuestro trabajo es llevar a cabo la implementación de las operaciones detectadas para la simplificación de expresiones numéricas en español, como una categoría específica de expresiones léxicas. Los resultados de los dos análisis realizados se usarán para esta implementación, considerando que algunas expresiones numéricas podrían ser eliminadas dependiendo del contexto y otras sustituidas para hacerlas más accesibles. Para ello tenemos la intención de llevar a cabo un análisis de los datos más profundo y detallado sobre un corpus extenso y obtener así un conjunto de reglas de transformación considerando además las necesidades del usuario final.

Agradecimientos

Queremos agradecer al Dr. Stefan Bott por su ayuda ofrecida con el manejo del analizador Freeling para realizar este trabajo.

Este trabajo ha sido parcialmente financiado por el Gobierno Español a través del Ministerio de Educación y Ciencia (TIN2009-14659-C03-01 Proyecto), Universidad Complutense de Madrid y Banco Santander Central Hispano (GR58/08 Beca de grupo de investigación) y el programa de becas de Formación de Personal de Investigación (FPI).

Este trabajo, en parte, ha sido realizado bajo el proyecto titulado Simplext: un sistema automático para simplificación de textos (Simplext: An automatic system for text simplification), con el número TSI-020302-2010-84 [17]. También queremos agradecer a la financiación del Programa Ramón y Cajal 2009 (RYC-2009-04291), Ministerio de Economía y Competitividad, Secretaría de Estado de Investigación, Desarrollo e Innovación de España.

Bibliografía

Anula, A. 2007. Tipos de textos, complejidad lingüística y facilicitación lectora. En *Actas del Sexto Congreso de Hispanistas de Asia*, páginas 45–61.

Anula, A. 2008. Lecturas adaptadas a la enseñanza del español como L2: variables lingüísticas para la determinación del nivel de legibilidad. En *La evaluación en el aprendizaje y la enseñanza del español como LE/L2, Pastor y Roca (eds.),*, páginas 162–170, Alicante.

Aswani, N., V. Tablan, K. Bontcheva, y H. Cunningham. 2005. Indexing and Querying Linguistic Metadata and Document Content. En *Proceedings of Fifth International Conference on Recent Advances in Natural Language Processing*, Borovets, Bulgaria.

Bautista, S., P. Gervás, y R.I. Madrid. 2009. Feasibility analysis for semiautomatic conversion of text to improve readability. En *The Second International Conference on Information and Communication Technologies and Accessibility*, May 2009.

Bautista, S., R. Hervás, P. Gervás, R. Power, y S. Williams. 2011. How to Make Numerical Information Accessible: Experimental Identification of Simplification Strategies. En *Conference on Human-Computer Interaction*, Lisbon, Portugal.

Biran, O., S. Brody, y N. Elhadad. 2011. Putting it Simply: a Context-Aware Approach to Lexical Simplificaion. En *Proceedings of the ACL.*

[17] http://www.simplext.es [Último acceso: 20/11/2012]

Bott, S. y H. Saggion. 2011. An Unsupervised Alignment Algorithm for Text Simplification Corpus Construction. En *Workshop on Monolingual Text-to-Text Generation*, Portland, USA, June. ACL.

Burstein, J., J. Shore, J. Sabatini, Yong-Won Lee, y M. Ventura. 2007. The automated text adaptation tool. En Candace L. Sidner Tanja Schultz Matthew Stone, y ChengXiang Zhai, editores, *HLT-NAACL (Demonstrations)*, páginas 3–4. The Association for Computational Linguistics.

Candido, Jr., A., E. Maziero, C. Gasperin, Thiago. A. S. Pardo, L. Specia, y Sandra M. Aluisio. 2009. Supporting the adaptation of texts for poor literacy readers: a text simplification editor for brazilian portuguese. En *Proceedings of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications*, páginas 34–42, Stroudsburg, PA, USA. Association for Computational Linguistics.

Carroll, J., G. Minnen, Y. Canning, S. Devlin, y J. Tait. 1998. Practical Simplification of English Newspaper Text to Assist Aphasic Readers. En *Proceedings of AAAI-98 Workshop on Integrating Artificial Intelligence and Assistive Technology*, páginas 7–10, Madison, Wisconsin.

Chandrasekar, Raman, Christine Doran, y Bangalore Srinivas. 1996. Motivations and Methods for Text Simplification. En *COLING*, páginas 1041–1044.

De Belder, J., K. Deschacht, y Marie-Francine Moens. 2010. Lexical simpli?cation. En *Proceedings of Itec2010 : 1st International Conference on Interdisciplinary Research on Technology, Education and Communication.*

Deschacht, Koen y Marie-Francine Moens. 2009. Semi-supervised semantic role labeling using the latent words language model. En *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, páginas 21–29, Stroudsburg, PA, USA. Association for Computational Linguistics.

Devlin, S. y G. Unthank. 2006. Helping aphasic people process online information. En *Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility*, Assets '06, páginas 225–226, New York, NY, USA.

Dieckmann, Nathan F., Paul Slovic, y Ellen M. Peters. 2009. The use of narrative evidence and explicit likelihood by decision-makers varying in numeracy. *Risk Analysis*, 29(10).

Drndarević, Biljana y Horacio Saggion. 2012. Reducing text complexity through automatic lexical simplificacion: an empirical study for spanish. *Procesamiento del Lenguaje Natural.*

Inui, K., A. Fujita, T. Takahashi, R. Iida, y T. Iwakura. 2003. Text simplification for reading assistance: A project note. En *In Proceedings of the 2nd International Workshop on Paraphrasing: Paraphrase Acquisition and Applications*, páginas 9–16.

Klebanov, B. B., K. Knight, y D. Marcu. 2004. Text simplification for information-seeking applications. En *On the Move to Meaningful Internet Systems, Lecture Notes in Computer Science*, páginas 735–747.

Lal, P. y S. Ruger. 2002. Extract-based summarization with simplification. En *Proceedings of the ACL 2002 Automatic Summarization.*

Maynard, D., V. Tablan, H. Cunningham, C. Ursu, H. Saggion, K. Bontcheva, y Y. Wilks. 2002. Architectural Elements of Language Engineering Robustness. *Journal of Natural Language Engineering – Special Issue on Robust Methods in Analysis of Natural Language Data*, 8(2/3):257–274.

Medero, J. y M. Ostendorf. 2011. Identifying targets for syntactic simplification. En *In Proceedings of the Workshop on Speech and Language Technology in Education.*

Mishra H, Mishra A, Shiv B. 2011. In praise of vagueness: malleability of vague information as a performance booster. *Psychological Science*, 22(6):733–8, April.

Padró, Ll., M. Collado, S. Reese, M. Lloberes, y I. Castelln. 2010. Freeling 2.1: Five years of open-source language processing tools. En *Proceedings of the Seventh International Conference on Language Resources and Evaluation*, Valletta, Malta.

Peters, Ellen, Judith Hibbard, Paul Slovic, y Nathan Dieckmann. 2007. Numeracy skill and the communication, comprehension, and use of risk-benefit information. *Health Affairs*, 26(3):741–748.

Power, Richard y Sandra Williams. 2012. Generating numerical approximations. *Computational Linguistics*, 38(1).

Qualifications y Curriculum Authority. 2010. Annual report and accounts. Informe técnico, Financial statements.

Quinlan, P. 1992. *The Oxford Psycholinguistic Database*. Oxford University Press.

Saggion, Horacio, Elena Gómez-Martínez, Alberto Anula, Lorena Bourg, y Estaban Etayo. 2011. Text simplification in simplext: Making texts more accessible. En *Proceedings of the Sociedad Española del Procesamiento del Lenguaje Natural*.

Siddharthan, Advaith. 2002. Resolving Attachment and Clause Boundary Amgiguities for Simplifying Relative Clause Constructs. En *Proceedings of the Student Research Workshop, 40th Meeting of the Association for Computacional Linguistics*.

Specia, L. 2010. Translating from Complex to Simplified Sentences. En *9th International Conference on Computational Processing of the Portuguese Language*, páginas 30–39.

Yatskar, M., Pang B., C. Danescu-Niculescu-Mizil, y L. Lee. 2010. For the sake of simplicity: Unsupervised extraction of lexical simplifications from wikipedia. *CoRR*, abs/1008.1986.

Apresentação de Projectos

Bifid: un alineador de corpus paralelo a nivel de documento, oración y vocabulario

Rogelio Nazar
Institut Universitari de Lingüística Aplicada
Universitat Pompeu Fabra
rogelio.nazar@upf.edu

Resumen

Este artículo presenta un algoritmo que integra distintos aspectos del procesamiento de corpus paralelo y que ha sido implementado como una aplicación web. El trabajo se enmarca en la lingüística computacional pero puede interesar a terminólogos, traductores y estudiantes de lenguas extranjeras. El sistema está diseñado para operar con cualquier par de lenguas ya que es exclusivamente estadístico. Acepta como entrada un corpus paralelo definido como un conjunto de documentos en una lengua A y sus traducciones en una lengua B. Sin requerir más especificaciones, el sistema puede separar el conjunto de documentos en las dos lenguas, alinear cada documento con su traducción y luego alinear los segmentos dentro de cada par de documentos para producir finalmente un vocabulario bilingüe que incluye unidades poliléxicas.

Palabras clave

Alineación de corpus paralelo, extracción de vocabularios bilingües, lexicografía computacional

Abstract

This paper presents an algorithm that integrates different aspects of parallel corpus processing, which is now implemented as a web application. It is a computational linguistics project but can also be of interest to translators, terminologists and foreign language learners. The system is designed to operate with any pair of languages since it is exclusively based on statistical techniques. It takes a parallel corpus as input, defined as a set of documents in a language A and their translations into a language B. Without any further specification, the system separates the set of documents in the two languages, aligns each document with its translation and then aligns the segments within each pair of documents to finally produce a bilingual vocabulary that includes multiword units.

Keywords

Bilingual lexicon acquisition, computational lexicography, parallel corpus alignment

1 Introducción

Si bien en los últimos años ha aparecido una gran cantidad de publicaciones en las que se describen metodologías para la obtención de terminología bilingüe desde corpus comparables (Gaussier et al., 2004; Daille y Morin, 2005; Morin et al., 2008) e incluso corpus no relacionados (Fung, 1995; Rapp, 1999; Nazar, Wanner, y Vivaldi, 2008), el procesamiento de corpus paralelo continúa siendo el método más consolidado para obtener vocabularios bilingües o como herramienta de apoyo a la traducción, particularmente en el caso de la traducción técnica o especializada (Kübler, 2011). Esta tendencia se ha visto sin duda favorecida por la masificación de la World Wide Web, la principal fuente de corpus paralelos en nuestra época (Almeida, Simões, y Castro, 2002; Resnik y Smith, 2003).

El presente artículo describe el sistema Bifid, un proyecto en curso de herramienta de alineación de corpus paralelo. Se implementa como aplicación web y opera de forma independiente de lengua, llevando a cabo los siguientes pasos en forma secuencial: 1) separar el corpus en las dos lenguas que lo componen; 2) alinear a nivel de documento; 3) alinear a nivel de oración; 4) extraer un vocabulario bilingüe con unidades poliléxicas; 5) comenzar de nuevo el proceso desde el paso 2 introduciendo como parámetro adicional el resultado obtenido en el paso 4.

El sistema se encuentra actualmente implementado como una demo online en forma de CGI Perl[1], pero debe advertirse que el propósito de este demostrador no es el de funcionar ya como un producto informático sino el de permitir al lector reproducir el experimento con otro corpus. El artículo no incluye capturas de pantalla ni instrucciones de uso porque no pretende poner el foco de atención en un programa informático concreto sino en la metodología que se

[1] La dirección URL del proyecto es la siguiente: http://www.bifidalign.com/

propone para resolver el problema de la alineación, lo cual representa un nivel mayor de abstracción. En un programa informático, por ejemplo, se busca el mejor rendimiento posible, por lo que no habría razón para no incorporar conocimiento explícito de las lenguas analizadas (en la forma de diccionarios, lematizadores, analizadores morfosintácticos, etc.). En el caso de esta propuesta, en cambio, lo que se pretende es averiguar qué resultado es posible conseguir sin la ayuda de estos recursos.

Además de obedecer al principio de parsimonia, el enfoque "pobre" en conocimiento lingüístico tiene una doble motivación, teórica y práctica. Desde un punto de vista teórico, se puede afirmar que un algoritmo capaz de resolver el problema de la alineación sin recurrir a conocimiento explícito sobre el par de lenguas analizadas tiene un mayor poder de generalización y permite poner de relieve fenómenos que trascienden las lenguas particulares. Desde el punto de vista práctico, es útil porque la gran mayoría de las lenguas del mundo no goza de los recursos lingüísticos comunes a las lenguas mayoritarias, y la adaptación de los recursos lingüísticos a una lengua minoritaria, incluso los de nivel más elemental como el etiquetado morfosintáctico, implica un coste en tiempo, esfuerzo y presupuesto que en muchos casos resulta inasumible. También desde el punto de vista práctico, la capacidad de este algoritmo para adaptarse a distintos tipos de datos facilita su aplicación a una diversidad de escenarios. Por mencionar un ejemplo, organismos internacionales como la Organización de las Naciones Unidas, el Fondo Monetario Internacional o la Unión Europea, por nombrar algunos de los más importantes, disponen de cantidades ingentes de documentos traducidos a una gran diversidad de lenguas y, si todo ese material se pudiera reaprovechar para la generación de recursos léxicos o terminológicos sin tener que analizar las particularidades de cada una de esas lenguas o la codificación específica que cada organismo aplica a sus documentos, estaríamos ante la posibilidad de producir recursos lexicográficos o terminológicos de gran envergadura y con presupuesto mínimo.

El artículo se organiza de la siguiente manera: la sección 2 presenta un breve repaso de los trabajos más importantes realizados en el área de la alineación de corpus paralelos en los distintos niveles. La sección 3 presenta los detalles del presente algoritmo en cada uno de los pasos sucesivos de la alineación. La sección 4 presenta una evaluación de los resultados de la aplicación en distintos corpus. La sección 5 describe cómo funciona la implementación en forma de aplicación web y, finalmente, la sección 6 enuncia las conclusiones de este trabajo.

2 Antecedentes de la alineación de corpus paralelo

Como señala Véronis (2000), el primer gran hito en la historia de la alineación de corpus paralelo se da en 1822 con el desciframiento de la Piedra Rosetta llevado a cabo por Jean François Champollion. Algo más de un siglo más tarde, en el trabajo de Weaver (1955) se puede comprobar que existe lo esencial de la idea, aunque sea de forma embrionaria, cuando describe los primeros métodos estadísticos de traducción automática. A pesar de este aporte visionario, la alineación de corpus paralelo no prosperó hasta después de la segunda mitad de la década de los años ochenta, cuando las computadoras fueron capaces de procesar grandes matrices numéricas. Hasta entonces, la mentalidad que primaba en lo que se conocía como traducción automática era ajena a los corpus paralelos o a los métodos estadísticos en general. El pensamiento normal de la época era el de los sistemas basados en reglas, como por ejemplo en el proyecto llamado justamente Rosetta Stone (Appelo y Landsbergen, 1986).

A comienzos de los años noventa, sin embargo, se produce la explosión de publicaciones sobre alineación de corpus paralelos con métodos estadísticos y el ámbito se constituye como un campo de estudio específico, diferenciado respecto de la traducción automática. Los primeros trabajos se centraron en la alineación de oraciones en función de su extensión medida en caracteres o palabras (Gale y Church, 1991a; Brown, Lai, y Mercer, 1991), bajo el supuesto de que existe una correlación entre la extensión de las oraciones del texto de origen con las del texto meta. Al mismo tiempo, se exploró también la posibilidad de extraer vocabularios bilingües mediante el cálculo de la coocurrencia de las unidades léxicas en los pares de oraciones ya alineadas (Gale y Church, 1991b).

Posteriormente se incorporaron nuevas pistas, como la detección de cognados como señaladores de la correspondencia entre oraciones (Church, 1993; Simard, Foster e Isabelle, 1993; McEnery y Oakes, 1995). Se estudió también la posibilidad de establecer una retroalimentación entre las salidas de los dos procesos de alineación oracional y a nivel de vocabulario, según la idea de que el vocabulario bilingüe que resulta de la alineación oracional puede servir para refinar la misma alineación oracional, creando así un círculo virtuoso (Kay y Röscheisen, 1993; Moore, 2002; Braune y

Fraser, 2010). Posiblemente el trabajo más influyente sea el de Brown et al. (1993), ya que en él se inspiran algunos de los alineadores más conocidos en la actualidad, como Hunalign (Varga et al., 2005), Giza++ (Och y Ney, 2000; Och y Ney, 2003) o Champollion (Ma, 2006), entre otros.

Algunas propuestas de alineadores incorporan conocimiento lingüístico explícito, ya sea en la forma de vocabularios bilingües (Hofland y Johansson, 1998) o información morfosintáctica (De Yzaguirre et al., 2000; Gammallo, 2005; Gómez Guinovart y Simões, 2009), pero prevalecen las visiones puramente estadísticas o incluso geométricas, como en el caso de Melamed (2000), que representa los textos paralelos en un plano (dos ejes) de manera que la mejor alineación entre las oraciones se selecciona calculando la distancia que tienen con la diagonal.

Queda aún trabajo por hacer en la integración de las distintas fases de alineación, y es sobre todo en ese sentido en el que este artículo presenta una nueva contribución. Han aparecido diversas herramientas que integran diferentes niveles de alineación, como Twente Aligner (Hiemstra, 1998), NATools (Simões y Almeida, 2003), o Uplug (Tiedemann, 2006), pero hasta la fecha no se había intentado un enfoque que integrara la totalidad de los procesos, como en una herramienta que, partiendo de cero, sin ningún tipo de información sobre las lenguas del corpus ni intervención humana, produjera resultados de alineación en todos los niveles, desde el documento hasta el léxico.

3 El algoritmo

El proceso de este algoritmo comienza por un conjunto de documentos escritos en dos lenguas desconocidas y la tarea consiste en separar los documentos en estas dos lenguas (subsección 3.1.), alinear cada documento con su correspondiente traducción u original (subsección 3.2.), alinear las oraciones del texto de una lengua con las oraciones del texto en la otra lengua (subsección 3.3.) y, finalmente, extraer un vocabulario bilingüe (subsección 3.4.).

En todos estos pasos la intervención humana es posible, ya que como el resultado del proceso descrito en cada subsección alimenta el proceso siguiente, un usuario siempre puede controlar y corregir eventuales errores producidos en cada paso con el objeto de mejorar el resultado posterior. En ningún caso, sin embargo, esta intervención humana es contemplada como una condición *sine qua non* para la tarea del algoritmo ni se ha tenido en cuenta esta posibilidad en la evaluación de los resultados descrita en la sección 4.

3.1 Separación de los documentos en lenguas

Partiendo de un conjunto de documentos escritos en una lengua con su correspondiente traducción a otra, la primera operación consiste en separar los documentos en dos subconjuntos correspondientes a cada una de las lenguas. Para ello, en esta operación el algoritmo asume la universalidad de la distribución de frecuencias del vocabulario en los textos. De acuerdo con el sencillo principio de que todos los documentos escritos en una misma lengua tendrán al menos una parte del vocabulario en común consistente en las unidades léxicas más frecuentes, podemos agrupar los documentos en función de la similitud que tengan con respecto a las n palabras más frecuentes de cada documento. Para ello se llevan a cabo los siguientes dos pasos:

1. Ordenar por frecuencia decreciente el vocabulario del documento más extenso del corpus, al que llamaremos documento D_a.
2. Ubicar en un conjunto A todos los documentos del corpus que entre sus 10 palabras más frecuentes tengan al menos 3 palabras en común con las 10 más frecuentes del documento D_a.

Si esta operación consigue dividir el corpus en las dos lenguas, ya es posible pasar a la fase siguiente. Si, en cambio, todos los documentos han quedado en un mismo conjunto, entonces esto quiere decir que estamos trabajando con lenguas muy similares. En tal caso, se asume que las lenguas de los documentos se distribuyen en mitades iguales del corpus, lo cual sería de esperar en un corpus paralelo. Para ello reutilizamos el rango dado a los documentos por la puntuación otorgada en el cálculo recién descrito, es decir, en la similitud que tienen con el documento más largo calculada en función de las palabras altamente frecuentes que tienen en común.

3.2 Alineación a nivel de documento

Tomando el resultado del proceso anterior, en este paso se construirá una tabla de correspondencias entre los documentos[2] de la lengua A con los documentos de la lengua B, que representa los coeficientes aplicados a cada una de las parejas de documentos obtenidas por el producto cartesiano

[2]Para los fines de este proceso, no se tiene interés en saber cuál documento es el original y cuál la traducción.

de ambos conjuntos. Las variables que aparecen en el conjunto de coeficientes (1) son definidas a continuación en esta sección.

$$C = \{l, ln, sim, voc, num, bvoc\} \quad (1)$$

$$w(i,j) = \prod_{n=1}^{|c|} (1 + c_n(i,j)) \quad (2)$$

Una vez calculados los coeficientes, cada pareja de documentos *i–j* recibe una puntuación final que es el producto de los coeficientes (2), a los que sumamos 1 para no perder toda la puntuación en caso de que con alguno de los coeficientes se obtenga un valor 0. Esta puntuación permite ordenar todas las parejas de documentos de mayor a menor y así ir "sacándolas" una a una. Por una decisión metodológica, no se permite a un mismo documento estar en más de dos parejas, sólo se le permite estar en la que tiene el valor más alto, pero esto se podría considerar un parámetro de ejecución. Una vez explicada la lógica general del proceso de selección, pasamos a definir cada uno de los coeficientes.

Coeficiente *l*: Este coeficiente simplemente compara el tamaño de los documentos en número de caracteres. Se fundamenta en un criterio similar al que utilizan Gale y Church (1991a) para la alineación a nivel oracional, bajo el supuesto de que el documento original y su traducción deben tener un tamaño similar, tal como se define en la ecuación 1, donde la expresión *length* refiere a la extensión en caracteres de un documento.

$$l(i,j) = \frac{\text{mín}(length(i), length(j))}{\text{máx}(length(i), length(j))} \quad (3)$$

Como una forma de atender a las diferencias de tamaño que se pueden producir como consecuencia de la distinta redundancia natural de una y otra lengua, fenómeno conocido como *Language Proportion Coefficient* o *LPC* (Choueka, Conley, y Dagan, 2000), se lleva a valor 1 todo par de documentos que al ser comparados arrojen un valor de similitud superior o igual a 0.7 (o cualquier otro umbral arbitrario que se ajuste como parámetro en la implementación).

Coeficiente *ln*: Aunque también está basado en la comparación de extensión en caracteres, a diferencia del anterior este otro coeficiente recurre a un criterio metatextual que es comparar el largo de los nombres de los documentos, suponiendo que las parejas de documentos original-traducción tendrán nombres de extensión similar. Este coeficiente queda sin efecto en los experimentos descritos en la sección 4 ya que los nombres de los ficheros que designan los documentos obedecen a códigos que no guardan relación con el contenido. Se define de la misma forma que el coeficiente anterior (ecuación 3), por lo tanto puede decirse que se trata del mismo coeficiente pero aplicado de forma distinta.

Coeficiente *sim*: Este coeficiente, al igual que el anterior, analiza el nombre de los documentos. Su función es tratar de encontrar una similitud ortográfica entre los nombres de los ficheros de una y otra lengua, bajo el supuesto de que los nombres del fichero original y su traducción pueden tener elementos en común en un nivel inferior a la palabra.

Tal como se mencionó en la sección 2, la idea de la aplicación de coeficientes de similitud ortográfica para la alineación de corpus paralelos ha sido utilizada ya con el objeto de encontrar cognados (por ejemplo, en McEnery & Oakes, 1995). El coeficiente aplicado en el presente algoritmo compara formas como vectores binarios cuyas dimensiones son bigramas de caracteres (secuencias de dos letras) y la similitud entre vectores se calcula utilizando el coeficiente de Dice, expuesto en la ecuación 4.

$$sim(I,J) = \frac{2|I \cap J|}{|I| + |J|} \quad (4)$$

Tanto este coeficiente como el anterior, al estar aplicados según criterios metatextuales, son de menor interés teórico. Sin embargo, en casos reales es muy frecuente encontrar este tipo de similitud ortográfica entre nombres de ficheros y, por lo tanto, el coeficiente puede tener un potencial práctico importante de cara a un usuario final del sistema. Como en el caso anterior, en el experimento llevado a cabo en este artículo (sección 4) este coeficiente también queda sin efecto ya que, por las particularidades de los corpus utilizados en los experimentos, los nombres de los ficheros están compuestos por símbolos arbitrarios que no guardan relación con el contenido de los textos.

Coeficiente *voc*: Como el primero de los coeficientes, este mide características propias de los documentos. Se fundamenta en la probabilidad de que una pareja de documentos original-traducción tenga elementos del vocabulario en común. Uno puede pensar, por ejemplo, en diversos símbolos, siglas y nombres propios que puedan escribirse de la misma forma aunque se trate de lenguas distintas. El coeficiente, definido en la ecuación 5, normaliza la cantidad de unidades del vocabulario en común por la cantidad de unidades de vocabulario distintas encontradas en el

más extenso de los dos documentos.

$$voc(I, J) = \frac{|I \cap J|}{\text{máx}(|I|, |J|)} \quad (5)$$

Coeficiente *num*: El coeficiente *num* funciona y se define de la misma manera que el coeficiente *voc*, solamente que se restringe a la detección de números, y el objeto de mantenerlos como coeficientes distintos tiene una utilidad práctica ya que de esta manera es más grande su contribución al peso final de la comparación.

Coeficiente *bvoc*: Este último coeficiente es opcional, ya que hace referencia a la aplicación de un vocabulario bilingüe tal como el que resulta del producto final del algoritmo, y que puede ser incorporado de nuevo a una segunda ejecución. Cualquier léxico bilingüe puede servir a este coeficiente pero, naturalmente, su calidad afectará la precisión del resultado final. La forma en que se calcula este coeficiente es exactamente igual a la que se expone en la ecuación 5, solo que esta vez, en lugar de comparar las mismas palabras, se comparan palabras equivalentes.

Es evidente cuál puede ser el servicio que puede prestar un vocabulario bilingüe para la alineación a nivel de documento: aquel par de documentos que contenga más palabras equivalentes será probablemente la alineación correcta.

3.3 Alineación a nivel de oración

En la alineación a nivel de oración, el algoritmo elabora una matriz similar a la descrita en la subsección anterior. Presupone, sin embargo, la existencia de una división en el texto (oraciones o segmentos de otra extensión) mediante el carácter de final de línea, un signo universal, presente por definición en todo archivo de texto. La situación en la que se encuentra el algoritmo en la alineación del corpus a nivel de la oración conserva una serie de similitudes con la alineación a nivel de documento, pero dispone de nuevas pistas, como la información posicional de las oraciones, definida a continuación:

Coeficiente *pos*: No era posible, en la subsección 3.2., hacer alguna suposición respecto a la situación posicional de los documentos en el corpus. A nivel oracional, en cambio, es legítimo suponer que existirá un orden que debe ser respetado al menos en parte por quien hizo la traducción a la otra lengua. Por lo tanto, es altamente probable que la primera oración del documento original se corresponda con la primera oración de la traducción y, de la misma forma, que la última oración del original se corresponda también con la última oración de la traducción. De esta manera, se observará una correlación entre la posición de cada oración en el original con respecto a la posición de su traducción en el texto meta. Esta correlación permite definir un coeficiente posicional tal como se indica en la ecuación 6, donde los símbolos $P_{i,a}$ y $P_{j,b}$ representan la posición relativa de cada oración a en el documento original i y la posición relativa de cada oración b en la traducción j.

$$pos(a, b) = \frac{\text{mín}(P_{i,a}, P_{j,b})}{\text{máx}(P_{i,a}, P_{j,b})} \quad (6)$$

El coeficiente *sim*, también incluido en la matriz para la alineación oracional, se aplica en este caso al contenido de las oraciones candidatas a alineación en lugar de al nombre de los documentos, tal como se hizo en la subsección 3.2., esta vez con el objeto de detectar la presencia de cognados. Funciona de forma paralela a otro coeficiente que mide los cognados, *cogn*, y la única diferencia es que en el caso de este último no se comparan las dos oraciones como una única cadena de caracteres sino que se comparan las palabras de las oraciones entre sí. La comparación se realiza de la misma manera que el coeficiente *sim* pero la lógica de su aplicación es ligeramente distinta. La idea sería que a mayor cantidad de cognados tenga un par de oraciones, más probable será que se trate de una alineación correcta. En este punto, además, la detección de números queda nuevo constituida como una variable independiente, con el coeficiente *num*, debido a la vital importancia que en algunos casos tienen los números para la alineación. Así, en la ecuación 7 queda definido un nuevo conjunto C prima de coeficientes.

$$C' = \{l, ln, pos, sim, num, cogn, voc, bvoc\} \quad (7)$$

La manera de seleccionar las mejores alineaciones entre oraciones es muy similar a la que se describe en la alineación a nivel de documento, con la diferencia de que en este caso se debe respetar el orden de las oraciones en los textos. El primer paso para la alineación es la búsqueda de puntos de anclaje a lo largo de los documentos alineados. Eso se consigue alineando primero todos aquellos pares de oraciones que sean más seguros. Concretamente, en esta implementación se toman como puntos de anclaje los pares de oraciones cuya puntación esté por encima del percentil 80, pero de nuevo esto puede funcionar como un parámetro más de la ejecución.

Una vez encontrados los puntos de anclaje resulta más fácil distribuir las oraciones dentro de

cada fragmento entre anclajes, aunque sin llegar a forzar una alineación uno-a-uno ya que la alineación oracional no es una función biyectiva. Con relativa frecuencia, una misma oración del texto meta puede alinearse con más de una oración en el texto original y viceversa, por lo tanto es necesario flexibilizar el criterio y permitir que una misma oración de un texto sea alineada con más de una oración de su contraparte.

3.4 Alineación a nivel de vocabulario

Una vez que el corpus ha sido separado en lenguas y alineado a nivel de documento y de oración, ya es posible elaborar una primera versión del vocabulario bilingüe, que podrá servir, posteriormente, como un parámetro para iterar el algoritmo. Es necesario advertir que no todos los autores consideran que la alineación a nivel de vocabulario equivale a la extracción de un vocabulario bilingüe, ya que en el primer caso se trata de alinear las unidades léxicas en el contexto mismo en el que ocurren (Santos y Simões, 2008). En este artículo se ha preferido la extracción del vocabulario bilingüe con independencia del contexto de aparición porque parece lo más útil desde el punto de vista práctico.

Tal como se advirtió ya en la introducción, el resultado de esta alineación a nivel de vocabulario no se limita únicamente a la palabra ortográfica, ya que la alineación a nivel de las unidades sintagmáticas es de una importancia capital en particular para usuarios interesados en la terminología especializada, que muy a menudo se presenta en forma de unidades poliléxicas. Por este motivo, tomamos como vocabulario para analizar no solamente las palabras ortográficas sino también todas las secuencias de hasta cinco palabras siempre y cuando no tengan como primer o último componente una palabra con menos de cuatro caracteres. La expansión del vocabulario no llega a saturar la memoria porque se descartan los hapax legomena y dis legomena.

$$C'' = \{l, sim, coo\} \tag{8}$$

Como en la alineación a nivel de documento y oración, en el caso de la alineación a nivel del léxico definimos nuevamente un conjunto biprima de coeficientes (8), en este caso con tres. Dos de ellos son comunes a las instancias anteriores. Se añade, sin embargo, un coeficiente de asociación basado en la coocurrencia de las palabras, el Coeficiente *coo*, descrito a continuación.

Coeficiente *coo*: Este coeficiente mide el grado de asociación estadística entre dos palabras a través de la coocurrencia en una misma alineación oracional. La gran mayoría de los autores que han realizado alineación de corpus paralelo a nivel de léxico han calculado de una forma u otra la coocurrencia de las unidades, utilizando para ello distintos coeficientes de asociación. El que se define en la ecuación 9 pone en relación la frecuencia de coocurrencia de las unidades léxicas i y j, candidatas a ser alineadas, normalizada por la frecuencia total de las unidades i y j en el corpus.

$$coo(i,j) = \frac{f(i,j)}{\sqrt{f(i).\sqrt{f(j)}}} \tag{9}$$

Tal como se describió antes en la ecuación 2, la puntuación final de cada pareja de unidades léxicas potencialmente equivalentes se calcula nuevamente como el producto de los coeficientes.

4 Evaluación de los resultados

Esta sección describe los resultados obtenidos con la aplicación de la metodología descrita en la sección 3 sobre tres corpus paralelos en los pares de lenguas inglés-castellano, inglés-francés y gallego-inglés. La subsección 4.1. describe las características de los corpus utilizados. Posteriormente, las subsecciones 4.2., 4.3., 4.4. y 4.5 describen, respectivamente, los resultados obtenidos durante las fases de reconocimiento de lengua, de alineación a nivel de documento, alineación a nivel de oración y alineación a nivel de vocabulario en cada uno de estos corpus. Es preciso tener en cuenta que en las subsecciones en las que se describen estos resultados aparecen distintos valores de precisión que corresponden a distintas ejecuciones del algoritmo, ya que, como se dijo en la introducción, se trata de un algoritmo iterativo: su resultado final es un vocabulario bilingüe que luego puede funcionar como un parámetro de entrada a una nueva ejecución, en un proceso de retroalimentación que le permite mejorar el desempeño. Este artículo describe los resultados obtenidos después de tres iteraciones, aunque no hay una razón teórica para limitar este número. De cualquier modo, es de esperar que la mejora en el desempeño sea menor con cada iteración.

4.1 Corpus utilizados en los experimentos

Con el objetivo de evaluar este sistema se llevaron a cabo experimentos de alineación en tres corpus paralelos diferentes. El primero es el corpus CLUVI-TECTRA, un conjunto de obras literarias en inglés con sus correspondientes traducciones al gallego (Gómez Guinovart y Sacau Fontenla, 2004), con un tamaño total aproximado de

CLUVI	JRC-ACQUIS	HANSARDS
100 %	95,4 %	100 %

Cuadro 1: Precisión de la separación de documentos por lengua

1.500.000 palabras. El segundo es una muestra del corpus JRC-Acquis (Steinberger et al., 2006) consistente en textos paralelos de la Unión Europea, de naturaleza legal. En el caso de este último corpus, se utilizaron sólo los documentos en castellano y en inglés del año 1990, lo que totaliza 800.000 palabras en las dos lenguas. El tercer corpus está constituido por una parte de las actas del parlamento canadiense (Canadian Hansards) del año 2000 publicadas por Ulrich Germann (2001), para el par de lenguas inglés-francés, también con un total de 800.000 palabras. El corpus CLUVI se divide en 600 ficheros, el JRC-Acquis en 332 y el Hansards en 42. Naturalmente, antes de utilizar estos corpus para la evaluación se eliminó toda la metainformación contenida en las etiquetas XML, dejando todos los documentos como texto plano.

4.2 Resultado de la separación de documentos por lengua

La precisión del resultado de la división del corpus por lenguas aparece detallada en la tabla 1 para los distintos corpus que han servido para el experimento. En este caso se expresan los resultados en una única vez ya que estos no se modifican con las distintas iteraciones como en los demás procesos de alineación. La discriminación por lengua resulta 100 % correcta excepto en el caso del corpus JRC-Acquis, y una de las razones que pueden explicar los errores es que los documentos en castellano de este corpus incluyen también largas leyendas en inglés, y cuando los documentos son muy cortos, lógicamente esta leyenda puede dificultar la detección.

4.3 Resultado de la alineación a nivel de documento

El cuadro 2 expone los resultados de la alineación a nivel de documento para cada uno de los corpus. Debido a que la alineación es afectada por las sucesivas iteraciones del algoritmo, en este cuadro se expone el resultado de las mismas muestras en tres iteraciones sucesivas. Como se puede apreciar, también en este caso la alineación a nivel de documento es óptima en el caso del CLUVI y los Hansards pero no en el JRC-Acquis, como consecuencia de los errores cometidos por el algoritmo en la instancia anterior.

iteración	CLUVI	JRC-ACQUIS	HANSARDS
1º	98,3 %	89,7 %	100 %
2º	100 %	90,9 %	100 %
3º	100 %	90,9 %	100 %

Cuadro 2: Precisión de la alineación a nivel de documento

iteración	CLUVI	JRC-ACQUIS	HANSARDS
1º	85,0 %	97,4 %	93,1 %
2º	86,8 %	97,9 %	94,3 %
3º	86,8 %	98,3 %	94,5 %

Cuadro 3: Precisión de la alineación oracional

4.4 Resultado de la alineación a nivel de oración

Para llevar a cabo la evaluación de la alineación a nivel de oración hacemos un muestreo aleatorio estratificado de cinco documentos por corpus (dos en el caso del Hansards, ya que son documentos mucho más largos), con el objeto de obtener una muestra representativa de los documentos de distinto tamaño, ya que es de esperar que la alineación oracional sea más difícil en el caso de los documentos más largos. Como en el caso anterior, en el cuadro 3 también se expone el rendimiento de las alineaciones en tres iteraciones.

La evaluación se lleva a cabo revisando manualmente los documentos de la muestra, y el porcentaje simplemente representa la proporción entre alineaciones correctas e incorrectas. Para evaluar se toma como medida el segmento, que coincide en general con una oración pero puede ocurrir que contenga más de una. La forma de evaluar es simplemente controlar que a cada oración del texto de origen le corresponda una alineación correcta en el texto meta.

En la alineación a nivel oracional, el corpus CLUVI es el que ha mostrado el peor rendimiento, lo cual no es del todo sorprendente dada la naturaleza del corpus. Un corpus literario es siempre más complejo, porque los traductores sienten mayor libertad y los segmentos se encuentran menos estructurados. No es infrecuente que se sustraigan o se inserten segmentos, tal como se puede apreciar en el ejemplo del cuadro 4. Además, en un corpus literario existe menor cantidad de símbolos y cognados que son frecuentes en un corpus técnico y que ayudan a un alineador de estas características.

En el caso del corpus JRC-Acquis los errores de alineación son infrecuentes, lo cual se puede explicar por la extrema pulcritud con que se ha llevado a cabo la traducción. Las traducciones son muy similares al original, son pocos los casos

Original	**Traducción al gallego**
Then old Luce ordered another martini and told the bartender to make it a lot drier.	Logo o Luce pediu outro martini, e aínda máis seco.
Listen.	
How long you been going around with her, this sculpture babe?' I asked him.	-¿E canto tempo levas saíndo con esa escultura?
I was really interested.	
Did you know her when you were at Whooton?'	¿Coñecíala xa cando estabas en Whooton?
Hardly.	-¿Como a ía coñecer?
She just arrived in this country a few months ago.'	Hai só uns meses que chegou a este país.
She did?	-¿Si?
Where's she from?'	¿De onde é?
She happens to be from Shanghai.'	-Pois é de Shanghai.
No kidding!	-¿En serio?
She Chinese, for Chrissake?'	¿É chinesa?
Obviously.'	-Dende logo.
No kidding!	
Do you like that?	-¿E gústache iso?
Her being Chinese?'	¿Que sexa chinesa?

Cuadro 4: Ejemplo de alineación oracional (fragmento de "The Catcher in the Rye", de J.D. Salinger)

en los que una oración se traduce por más de una y solo en muy contadas ocasiones los traductores han eliminado o insertado pasajes. En el caso del Hansards la proporción de errores es ligeramente mayor, pero se trata también de una traducción sumamente fiel al original, se podría decir incluso "ideal" para una alineación.

Hay que reconocer que los usuarios del sistema aquí presentado no siempre utilizarán corpus paralelos de una calidad comparable a estos, lo cual es un factor de riesgo para la calidad del resultado. En cualquier caso, también hay que destacar que gracias a la estrategia de los puntos de anclaje, cuando se produce un error en una alineación no hay una reproducción en cadena de ese error, ya que rápidamente se recupera la alineación correcta en las oraciones siguientes. Además, los errores de alineación casi siempre se dan en oraciones contiguas, la oración correspondiente en la traducción está a una o dos posiciones antes o después de la que se seleccionó erróneamente.

4.5 Resultado de la alineación a nivel de léxico

El último paso de esta evaluación produce un vocabulario bilingüe que incluye unidades poliléxicas. Como en los casos previos, en esta subsección se exponen los resultados de tres iteraciones. Se evaluaron manualmente los primeros 2.000 pares de equivalentes resultantes de cada experimento. Naturalmente, es de esperar que la calidad de los resultados decaiga progresivamente al seguir elementos que aparecen más abajo en la lista. En el cuadro 5 se exponen los porcentajes de precisión como la proporción de alineaciones correctas para los tres corpus en las tres iteraciones. Tal como se puede apreciar, en

iteración	CLUVI	JRC-ACQUIS	HANSARDS
1º	99,8 %	96,8 %	97,8 %
2º	99,9 %	98,4 %	98,6 %
3º	99,9 %	98,4 %	98,8 %

Cuadro 5: Resultados de la alineación a nivel de vocabulario

todos los casos el aumento de la precisión de alineaciones se da fundamentalmente de la primera ejecución a la segunda. Prácticamente no hay diferencias entre la segunda y la tercera ejecución en el caso de los 2.000 pares mejor posicionados. El cuadro 6 muestra algunos ejemplos de la alineación léxica en el corpus JRC-Acquis, en las posiciones 357-371 de la lista. Como se puede apreciar, el algoritmo es capaz de resolver alineaciones léxicas complejas como las de unidades de tres componentes (*microorganismos modificados genéticamente*) con otro de cuatro componentes (*genetically modified micro organisms*).

En cuanto a los errores que se producen en la alineación a nivel de vocabulario, estos se dan casi exclusivamente en el caso de la alineación de términos poliléxicos, y se perciben en mayor proporción en el caso del corpus JRC-Acquis debido a que allí los términos poliléxicos son mucho más frecuentes por la naturaleza más técnica de ese corpus. Los siguientes son algunos ejemplos de alineaciones incorrectas:

- *consecutivos durante* ≠ *consecutive years during*
- *prueba suficiente* ≠ *being sufficient proof*
- *monetaria internacional* ≠ *competent international monetary*
- *república democrática* ≠ *German Democratic Republic*

Rango	Término castellano	Término inglés
...	...	...
357	autoridades administrativas	administrative authorities
358	provisionales	provisional
359	microorganismos modificados genéticamente	genetically modified micro organisms
360	iniciativa	initiative
361	expertos	experts
362	recomendaciones	recommendations
363	portugal	portugal
364	microorganismos	micro organisms
365	integrado	integrated
366	programa	programme
367	contacto	contact
368	interpretación uniforme	uniform interpretation
369	racional	rational
370	diferencia	difference
371	secretario general	secretary general
...	...	...

Cuadro 6: Ejemplos de alineación léxica en el corpus JRC-Acquis

Estos ejemplos, que son representativos de los errores que se encuentran, llevan a pensar que se podrían resolver con un mínimo grado de conocimiento lingüístico, tal como un modelo de sintaxis derivado de un etiquetador morfosintáctico, que se puede conseguir con facilidad en el caso de la mayoría de las lenguas europeas.

5 La interfaz

Esta sección ofrece una breve descripción del funcionamiento de la interfaz web que es la forma que se propone como implementación del algoritmo. Tal como se advierte en la introducción, el artículo no pretende ofrecer una descripción pormenorizada de los aspectos técnicos de la aplicación informática en sí, ya que el interés del trabajo está más en el método que en el programa.

El programa en sí es menos importante porque un programador podría preferir implementar el mismo algoritmo en C en lugar de Perl por cuestiones de eficiencia, o hacerlo como una aplicación de escritorio en lugar de una aplicación web. Los aspectos técnicos de un producto informático son complejos y motivarían un artículo diferente. El desarrollo de productos informáticos requiere estudios de usuario y el cuidado de una serie de aspectos relacionados con la usabilidad de las interfaces. Por ejemplo, la demo online solo acepta como entrada ficheros de texto plano. Un terminólogo o traductor no tiene por qué saber cuál es la diferencia entre un archivo binario y un archivo de texto, por tanto, si se le solicita un archivo de texto probablemente proporcionará un documento de Word o un PDF. Algo en apariencia trivial como la conversión de formatos PDF a texto puede resultar muy complejo en algunos casos (eliminación o tratamiento de símbolos, tablas, fórmulas y epígrafes de las figuras o imágenes, reconocimiento y conversión de codificación de caracteres, reconstrucción de texto dividido en columnas, de palabras que se cortan a final de línea, restauración de diacríticos, de errores de reconocimiento óptico de caracteres y toda una serie de temas que no tienen relación intrínseca con la alineación de corpus paralelo), por lo que se ha preferido dejar de lado ese tipo de cuestiones técnicas en favor de un modelo básico y una argumentación más abstracta.

Dicho esto, también es verdad que el mismo diseño de Bifid facilita su utilización por parte de usuarios no informáticos, lo cual no deja de ser importante ya que, si bien se han presentado distintas herramientas para la alineación de corpus paralelo, como se comentó en la sección 2, en general estas no se caracterizan por un diseño amigable para un usuario sin conocimientos avanzados de informática. Ejecutar una aplicación en línea de comando, como muchas de ellas requieren, no es algo que esté en el horizonte de posibilidades de la mayoría de los usuarios no expertos en informática en la actualidad. Es, por tanto, un valor práctico que el sistema tiene ya en su estado actual de desarrollo: el poder ser operado con facilidad por un usuario que al menos sea capaz de proporcionar un corpus en forma de archivos de texto.

La particularidad de una implementación de este algoritmo en forma de aplicación web tiene obvias ventajas como el poder ser ejecutada en cualquier plataforma sin necesidad de llevar a cabo una instalación. Sin embargo, la decisión también acarrea algunos inconvenientes, principalmente que el coste computacional del sistema y las limitaciones de infraestructura hacen que no sea posible una respuesta instantánea del servidor a las solicitudes de los usuarios. El diseño del algoritmo tiene aún que mejorar para funcio-

nar más rápido, pero por el momento la potencia del hardware es un factor clave, y ello exige que el programa funcione en varios servidores en red para que sea viable.

En este momento el sistema se encuentra instalado en un solo servidor, y en lugar de devolver los resultados de inmediato, envía una dirección URL al correo electrónico que el usuario indica en la solicitud. En esta URL es posible observar los resultados a medida que se van generando, pero se debe esperar que estos resultados "maduren" para obtener la calidad óptima, lo que se consigue después de dos o tres iteraciones. El envío de los datos se da por medio de un formulario web a través del cual el usuario sube un archivo comprimido que contiene los documentos de su corpus paralelo. Existe la posibilidad de realizar correcciones en la salida de cada proceso para evitar que los errores se propaguen a las instancias posteriores. A la salida de cada proceso el usuario puede descargar la información en ficheros de texto con nombres que corresponden a cada proceso ("lang.txt" para la separación de lenguas, "doc.txt" para la alineación a nivel de documento, etc.) que podrá modificar con un editor de texto para luego repetir el experimento incluyendo estos ficheros en el archivo del corpus.

En su estado actual, el tiempo de respuesta del sistema varía en función del tamaño del corpus y de la carga de usuarios, pero como referencia general, el procesamiento con tres iteraciones de un corpus de un millón de palabras puede tardar 24 horas, a lo que hay que sumar el tiempo que lleve el proceso en lista de espera. Una leyenda en la interfaz informa en todo momento sobre el número de trabajos pendientes.

6 Conclusiones

Este artículo ha presentado un sistema integral de alineación de corpus paralelo que no incorpora ningún tipo de conocimiento lingüístico y que puede por tanto ser utilizado en cualquier par de lenguas. La datos generados muestran que los resultados son competitivos y que ya es posible su aplicación a casos reales para obtener material de una calidad suficiente como para que sea rentable el posterior procesamiento y corrección por parte del usuario, tarea que puede ser llevada a cabo a lo largo de cada uno de los pasos del proceso (alineación a nivel de documento, de oración y de léxico).

En general, la calidad de los resultados tanto en el nivel de la alineación oracional como la del vocabulario, que son los dos niveles que han sido ya explorados en la bibliografía, se encuentran en el mismo orden que los mejores resultados de las publicaciones consultadas aunque, como es sabido, las comparaciones son siempre meramente orientativas por las diferentes características de los corpus y las lenguas analizadas por cada autor. Para conseguir una comparación rigurosa sería preciso llevar a cabo un experimento con distintos algoritmos trabajando sobre un mismo corpus, que se deja para trabajo futuro ya que no era el objetivo principal de este artículo.

Del diseño del algoritmo se puede decir que es original en lo teórico y económico en lo práctico, ya que permite una alta portabilidad a otras lenguas sin necesidad de organizar o procesar lingüísticamente el corpus. Hay que decir, con todo, que aún es necesario seguir experimentando con distintos pares de lenguas antes de poder afirmar categóricamente que es independiente de lengua, ya que, como algunos autores advierten (Choueka, Conley, y Dagan, 2000), estos algoritmos deben ser evaluados en pares como hebreo-inglés, árabe-inglés, chino-inglés, etc. En el caso de las lenguas altamente aglutinantes, como el turco, es de esperar que los resultados sean peores. Es también el caso del swahili, por ejemplo, en el que una secuencia en inglés como "I have turned him down" se puede traducir con una sola unidad léxica, "Nimemkatalia" (Pauw, Wagacha, y Schryver, 2009).

En cuanto a trabajo futuro, existe toda una serie de mejoras que se están llevando a cabo de cara a una nueva versión de este alineador que será de mayor complejidad. Estas mejoras son el objeto de un nuevo artículo que se encuentra en preparación y que se dedica por entero a medir cómo cambia la calidad de los resultados en cada uno de los niveles de alineación con cada modificación que se hace sobre la versión básica presentada en este artículo.

Agradecimientos

Este proyecto es posible gracias a un contrato del autor como técnico en el Institut Universitari de Lingüística Aplicada de la Universitat Pompeu Fabra. Agradezco a Alberto Simões, Alberto Lugrís y Diana Santos por leer una versión previa del artículo y contribuir con sus comentarios a mejorarlo sustancialmente.

Bibliografía

Almeida, J., A. Simões, y J. Castro. 2002. Grabbing parallel corpora from the web. *Procesamiento del Lenguaje Natural*, (29):13–20.

Appelo, L. y J. Landsbergen. 1986. The Machine Translation Project Rosetta. En *International Conference on the State of Machine Translation in America, Asia and Europe. Proceedings of IAI-MT86, 20-22 August*, Bürgerhaus, Dudweiler.

Braune, F. y A. Fraser. 2010. Improved unsupervised sentence alignment for symmetrical and asymmetrical parallel corpora. En *Proceedings of the 23th International Conference on Computational Linguistics Coling*, páginas 81–89.

Brown, P., V. DellaPietra, S. DellaPietra, y R. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19(2):263–311.

Brown, P.F., J. C. Lai, y R. L. Mercer. 1991. Aligning sentences in parallel corpora. En *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL)*, páginas 169–176, Berkeley.

Choueka, Y., E. Conley, y I. Dagan. 2000. A comprehensive bilingual word alignment system. application to disparate languages: Hebrew and English. En *Parallel Text Processing: Alignment and Use of Translation Corpora*. Kluwer, páginas 69–96.

Church, K.W. 1993. Charalign: a program for aligning parallel texts at the character level. En *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics (ACL)*, páginas 1–8, Columbus, Ohio.

Daille, B. y E. Morin. 2005. French-English terminology extraction from comparable corpora. En *Proceedings of the Second international joint conference on Natural Language Processing*, IJCNLP'05, páginas 707–718, Berlin, Heidelberg. Springer-Verlag.

De Yzaguirre, Ll., M. Ribas, J. Vivaldi, y M. T. Cabré. 2000. Some technical aspects about aligning near languages. En *Proceedings of the 2nd International Conference on Language Resources and Evaluation, (LREC'2000)*, páginas 545–548, Athens, Greece.

Fung, P. 1995. Compiling bilingual lexicon entries from a non-parallel English-Chinese corpus. En *Proceedings of the Third Workshop on Very Large Corpora*, páginas 173–183.

Gale, W. y K. Church. 1991b. Identifying word correspondence in parallel texts. En *Proceedings of the workshop on Speech and Natural Language*, HLT '91, páginas 152–157, Stroudsburg, PA, USA. Association for Computational Linguistics.

Gale, W. A. y K.W. Church. 1991a. A program for aligning sentences in bilingual corpora. En *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL)*, páginas 177–184, Berkeley.

Gammallo, P. 2005. Extraction of translation equivalents from parallel corpora using sense-sensitive context. En *Proceedings of Conference of the European Association for Machine Translation (EAMT'05)*, Budapest, Hungary.

Gaussier, E., J.M. Renders, I. Matveeva, C. Goutte, y H. Dejean. 2004. A geometric view on bilingual lexicon extraction from comparable corpora. En *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, páginas 526–533, Barcelona, Spain, July.

Germann, U. 2001. Aligned Hansards of the. 36th. Parliament of Canada - release 2001-1a. Informe técnico, http://www.isi.edu/natural-language/download/hansard/.

Gómez Guinovart, X. y E. Sacau Fontenla. 2004. Parallel corpora for the Galician language: building and processing of the CLUVI (Linguistic Corpus of the University of Vigo). En *Proceedings of the 4th International Conference on Language Resources and Evaluation, LREC 2004*, páginas 1179–1182, Lisboa (Portugal), May.

Gómez Guinovart, X. y A. Simões. 2009. Parallel corpus-based bilingual terminology extraction. En *Proceedings of the 8th International Conference on Terminology and Artificial Intelligence*, IRIT (Institut de recherche en Informatique de Toulouse), Université Paul Sabatier, Toulouse.

Hiemstra, D. 1998. Multilingual domain modeling in Twenty-One: automatic creation of a bi-directional translation lexicon from a parallel corpus. En *Computational linguistics in the Netherlands 1997*, volumen 25 de *Language and computers*, páginas 41–58, Amsterdam, the Netherlands. Rodopi.

Hofland, K. y S. Johansson. 1998. The translation corpus aligner: A program for automatic alignment of parallel texts. En *Corpora and Cross-linguistic research. Theory, Method, and Case Studies*. Rodopi, Amsterdam/Atlanta, páginas 87–100.

Kay, M. y M. Röscheisen. 1993. Text-translation alignment. *Computational Linguistics*, 19(1):121–142.

Kübler, N. 2011. Working with corpora for translation teaching in a French-speaking setting. En *New Trends in Corpora and Language Learning*. London, UK, páginas 62–80.

Ma, Xiaoyi. 2006. Champollion: A robust parallel text sentence aligner. En *Proceedings of the 5th International Conference on Language Resources and Evaluation, (LREC'2006)*, Genova, Italy.

McEnery, A. M. y M. P. Oakes. 1995. Sentence and word alignment in the CRATER project: methods and assessment. En *Proceedings of the EACL-SIGDAT Workshop: from texts to tags, Issues in Multilingual Language Analysis(ACL)*, páginas 77–86, Dublin, Ireland.

Melamed, D. 2000. Pattern recognition for mapping bitext correspondence. En *Parallel Text Processing: Alignment and Use of Translation Corpora*. Kluwer, páginas 25–47.

Moore, R. 2002. Fast and accurate sentence alignment of bilingual corpora. En *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas on Machine Translation: From Research to Real Users, AMTA '02*, páginas 135–144, London, UK. Springer-Verlag.

Morin, E., B. Daille, K. Takeuchi, y K. Kageura. 2008. Brains, not brawn: The use of "smart" comparable corpora in bilingual terminology mining. *ACM Trans. Speech Lang. Process.*, 7(1):1–23, Octubre.

Nazar, R., L. Wanner, y J. Vivaldi. 2008. Two step flow in bilingual lexicon extraction from unrelated corpora. En *Proceedings of the 12th conference of the European Association for Machine Translation*, páginas 138–147, Hamburg: HITEC.

Och, F. y H. Ney. 2000. Improved statistical alignment models. En *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, ACL '00, páginas 440–447, Stroudsburg, PA, USA. Association for Computational Linguistics.

Och, F. y H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Pauw, G. De, P. Wagacha, y G. De Schryver. 2009. The SAWA Corpus: a parallel corpus English-Swahili. En *Proceedings of the First Workshop on Language Technologies for African Languages (AfLaT 2009)*. Association for Computational Linguistics.

Rapp, R. 1999. Automatic identification of word translations from unrelated English and German corpora. En *Proceedings of 37th Annual Meeting of the ACL*, páginas 519–526.

Resnik, P. y N. Smith. 2003. The web as a parallel corpus. *Computational Linguistics*, 29(3):349–380, Septiembre.

Santos, D. y A. Simões. 2008. Portuguese-English word alignment: some experiments. En *Proceedings of LREC 2008 Workshop on Comparable Corpora*, páginas 2988–2995, Marrakech, Marroco.

Simões, A. y J. Almeida. 2003. Natools - a statistical word aligner workbench. *Procesamiento del Lenguaje Natural*, (31):217–226.

Simard, M., G. Foster, y P. Isabelle. 1993. Using cognates to align sentences in bilingual corpora. En *Proceedings of the 1993 conference of the Centre for Advanced Studies on Collaborative research: distributed computing - Volume 2*, CASCON '93, páginas 1071–1082. IBM Press.

Steinberger, R., B. Pouliquen, A. Widiger, C. Ignat, T. Erjavec, D. Tufis, y D. Varga. 2006. The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. En *In Proceedings of the 5th International Conference on Language Resources and Evaluation*.

Tiedemann, J. 2006. ISA & ICA - two web interfaces for interactive alignment of bitexts. En *Proceedings of the 5th International Conference on Language Resources and Evaluation, (LREC'2006)*, Genova, Italy.

Varga, D., L. Németh, P. Halácsy, A. Kornai, V. Trón, y V. Nagy. 2005. Parallel corpora for medium density languages. En *Proceedings of the RANLP 2005*, páginas 590–596.

Véronis, J. 2000. From the Rosetta stone to the information society: A survey of parallel text processing. En *Parallel Text Processing: Alignment and Use of Translation Corpora*. Kluwer, páginas 1–24.

Weaver, W. 1955. Translation. En *Machine Translation of Languages*. MIT Press, Cambridge, Massachusetts, páginas 15–23.

inLéctor: creación de libros electrónicos bilingües interactivos

Antoni Oliver
Universitat Oberta de Catalunya
aoliverg@uoc.edu

Miriam Abuin Castro
Universitat Oberta de Catalunya
mabuinc@uoc.edu

Resumen

En este artículo presentamos el proyecto InLéctor para la creación de libros electrónicos bilingües interactivos. El objetivo del proyecto es desarrollar una serie de aplicaciones para la creación automática de libros electrónicos bilingües. Dichos libros permiten pasar del texto original al traducido con un solo clic y se publican en los formatos de libros electrónicos más habituales: html, epub y mobi. La intención es publicar obras literarias en dominio público (cuyos derechos de autor hayan caducado) con traducciones en dominio público (los derechos de traducción también han caducado). Los programas desarrollados se basan en software libre y se publicarán también bajo una licencia libre. De esta manera las editoriales que lo deseen podrán publicar su catálogo en este formato con una inversión mínima. En un futuro se pretende incluir enlaces al audio correspondiente a la lectura humana de la obra original.

Palabras clave

libros electrónicos, traducción, alineación de textos

Abstract

In this paper we present the project InLéctor, aiming to create interactive bilingual ebooks. The goal of this project is to develop a set of applications for the automatic creation of bilingual ebooks. These books allow to switch from the original text to the translation with a simple click and are published in the more popular ebook formats: html, epub and mobi. We plan to publish literary works in the public domain with translations also in the public domain. All the developed applications are based on free software and will be also published under a free license. Publishing companies will be able to use this software at no cost, which will make possible to publish their catalogue in this format. At this moment we are publishing bilingual ebooks, but in the near future we plan to include links to the audio of the human reading of the original work.

Keywords

ebooks, translations, text alignment

1 Introducción

El proyecto InLéctor(Oliver, Coll-Florit, y Climent, 2012) pretende fomentar la lectura en versión original, ofreciendo libros bilingües, en texto y audio, en un entorno de lectura interactiva. En este proyecto pretendemos desarrollar una metodología para la creación automática de libros bilingües, utilizando software libre y publicando con licencia libre los programas desarrollados. Además, pondremos a la disposición del público gratuitamente una serie de obras literarias bilingües. Con esta iniciativa esperamos fomentar la publicación de obras literarias bilingües, en dos escenarios:

- Mediante la creación de una comunidad de usuarios que creen nuevas obras literarias en este formato. Las obras literarias que se creen deberán estar en dominio público, tanto el original como su traducción.
- Ofreciendo el software a las empresas editoriales para que puedan publicar su fondo bibliográfico en este formato con una inversión mínima.

Las obras se publicarán en los formatos más extendidos (html, epub y mobi) de manera que se puedan visualizar correctamente en los dispositivos de lectura más habituales

2 Funcionalidades

2.1 Texto bilingüe

Las obras se ofrecerán en su lengua original y en su traducción a otra lengua. El original y la traducción estarán paralelizados a nivel de oración. Esto permitirá un cambio rápido de la versión original a la traducida con un simple clic. De esta manera, si el lector quiere consultar la traducción de una determinada oración, haciendo clic sobre la oración original se visualizará la oración traducida en su contexto, es decir, dentro de la obra traducida. De esta manera, el lector

podrá continuar leyendo la traducción y pasar en cualquier momento al texto original si lo desea.

Si en algún caso no se ha obtenido una alineación de una determinada oración, haciendo clic sobre esta se irá a la oración traducida anterior más cercana. De esta manera nos aseguramos de que todas las oraciones tengan un enlace y el usuario apenas si notará un pequeño desplazamiento del texo.

Para el formato mobi, debido a restricciones de visualización de los enlaces, la alineación se hará a nivel de párrafo y mediante un asterisco entre corchetes ([*]) antes de cada párrafo.

2.2 Audio de la lectura humana del original

Está previsto enlazar el texto original con el audio correspondiente a la lectura humana de la obra en la lengua original. Este aspecto puede ser de gran interés para la mejora de la comprensión oral y la pronunciación. En este caso, la alineación entre el texto y el audio no se realizará a nivel de oración, si no por unidades más amplias aún por determinar (párrafos, conjunto de párrafos o capítulos).

2.3 Glosario interactivo

Antes de iniciar la lectura de un capítulo o fragmento de una obra, el usuario podrá especificar su nivel de lengua lo que le proporcionará un glosario de las palabras más difíciles del texto. El objetivo es que el usuario aprenda el significado de estas palabras más complicadas a priori para así poder disfrutar de una lectura más ágil y con menos interrupciones. Este glosario interactivo se podrá descargar desde la web del proyecto indicando la obra o capítulo y el nivel de lengua del usuario.

Para generar los diccionarios bilingües se utilizarán fuentes libres como por ejemplo Wiktionary[1], WordNets libres (Bond y Paik, 2012) y los diccionarios de transferencia de Apertium (Forcada, Tyers, y Ramírez, 2009).

También se generarán diccionarios bilingües y monolingües libres de manera que se puedan utilizar como diccionario por defecto en los diferentes dispositivos. De esta manera el usuario podrá consultar el significado o la definición de una palabra en cualquier momento. Si el dispositivo ya cuenta con diccionarios bilingües instalados para el par de lenguas en uso, estos podrán utilizarse con las obras de nuestro proyecto, ya que utilizan formatos estándar.

[1] `http://www.wiktionary.org/`

2.4 Lectura ampliada

Por *lectura ampliada* entendemos la aproximación a un texto con la ayuda de información adicional que estará disponible al lector mediante un solo clic. La información adicional puede ser de tipo enciclopédico, visual o sonoro. Algunos ejemplos pueden ser la información sobre un lugar geográfico o persona que aparece en el texto; la visualización de una obra de arte o paisaje y la reproducción de un fichero de audio musical relacionada con la obra.

Pongamos un ejemplo hipotético: el protagonista de la obra que estamos leyendo entra en un museo y observa una determinada obra de arte mientras que en el hilo musical se reproduce una determinada sinfonía. El sistema de lectura ampliada enlazaría directamente con la página web del museo, una imagen y explicación de la obra de arte y daría la posibilidad de escuchar la misma sinfonía mientras leemos el pasaje.

2.5 Interacción entre los usuarios

La web del proyecto incluirá una serie de funcionalidades que permitan la interacción de los usuarios. El sistema permitirá compartir comentarios sobre la obra o dudas sobre un determinado fragmento. Esta interacción se llevará a cabo mediante del uso de redes sociales, como Facebook o Twitter.

3 Obtención de obras literarias

Las obras que se publiquen dentro de este proyecto serán únicamente aquellas que tengan los derechos de autor y de traducción libres, es decir, que estén en dominio público. De esta manera, podemos garantizar que la distribución de las obras sea totalmente legal. Así, las obras literarias en versión original y las traducciones se extraerán principalmente de Wikisource[2] y del Proyecto Gutemberg[3]. Aunque la caducidad de los derechos de autor depende de cada legislación, de manera general se puede considerar que una obra es de dominio público si han pasado más de setenta años desde la muerte de su autor[4]. Concretamente en España la legislación dicta que las obras quedan en dominio público ochenta años después de la muerte del autor si éste murió antes del 7 de diciembre de 1987, y setenta años después de su muerte si este falleció después de

[2] `http://wikisource.org/`

[3] `http://www.gutenberg.org/`

[4] Se puede encontrar información muy detallada en `http://en.wikisource.org/wiki/Help:Public_domain`

la mencionada fecha.

En cuanto a los audios, serán en su mayoría provenientes de LibriVox[5], proyecto en el que un gran número de voluntarios leen capítulos de libros que están bajo dominio público, y donde se publican también bajo dominio público los ficheros de audio.

4 Software utilizado

4.1 Softare general

Para la creación de los libros electrónicos se necesita contar con una serie de software general que está disponible con licencia libre. Entre ellos es necesario contar con un buen editor de textos. Existen muchísimas opciones y dependrán del sistema operativo que se utilice. Algunos ejemplos son Jedit[6] (multiplataforma) o notepad++[7] (para Windows). Lo importante es que el editor elegido cuente con un buen soporte para la creación de macros. También puede ser útil la utilización de editores específicos para XML, como por ejemplo XMLCopyEditor[8], o herramientas de validación de XML como *xmllint*.

4.2 Entorno de programación

Los programas están desarrollándose en Python[9] en combinación con el *Natural Language Toolkit*[10] (NLTK) (Loper y Bird, 2002). NLTK es un conjunto de bibliotecas y programas para el Procesamiento del Lenguaje Natural. Mediante NLTK se pueden programar fácilmente las tareas más habituales de procesamiento del lenguaje. En nuestro proyecto lo estamos utilizando para la segmentación de las obras en oraciones.

4.3 Alineación automática

La alineación de los textos correspondientes a la obra original y traducido la realizamos con el alineador automático Hunaling (Varga et al., 2007). Este alineador nos permite obtener las relaciones entre la frase original y la frase traducida. No todas las frases originales obtienen alineaciones con una o más frases traducidas. Esto se puede deber a dos motivos: o bien la frase original no tiene una traducción o bien el programa no ha podido obtener una alineación válida.

Los resultados de la alineación automática mejoran notablemente si en lugar de alinear los textos en sí, alineamos una versión lematizada de los textos. De esta manera, el número de formas se reduce notablemente ya que todas las variantes morfológicas se reducen a un mismo lema. Si además proporcionamos diccionarios bilingües al programa de alineación, los resultados también pueden mejorar considerablemente.

Aunque no todas las oraciones originales obtengan una alineación válida, todas ellas tendrán un enlace a una oración correspondiente a la traducción. Este enlace será la oración traducida en caso de obtener alineación, o bien la oración traducida inmediatamente anterior que haya obtenido una alineación válida.

4.4 Creación de libros electrónicos

Existen una gran variedad de software libre para la creación de libros electrónicos. Hasta el momento hemos utilizado Calibre[11], ya que permite la creación de libros en formato epub y mobi, tanto desde una interfaz gráfica de usuario, como desde un terminal, lo que permite automatizar aún más el proceso.

Existen otras opciones para la creación de libros electrónicos, especialmente si se pretende utilizar el formato epub (que es el estándar libre). Una de ellas es Sigil[12], un editor gráfico para la creación de libros en formato epub. También debe destacarse python-epub-builder[13], un paquete de Python que permite desarrollar programas que automaticen totalmente la creación de libros en formato epub.

4.5 Procesado lingüístico

El procesado lingüístico necesario para la creación de libros paralelos se concreta en dos tareas: la segmentación en oraciones y la lematización.

La segmentación en oraciones es necesaria como paso previo a la alineación. Este paso es decisivo para lograr una buena alineación, y de ningún modo debe considerarse una tarea trivial. Hasta el momento estamos utilizando un segmentador genérico proporcionado por el NLTK. Probablemente, en futuras versiones del sistema, se tendrá que profundizar en este aspecto.

Para el lematizado de los textos se han utilizado etiquetadores morfosintácticos que nos proporcionan, para cada palabra, su lema y una etiqueta morfosintáctica. De esta información nos

[5] `http://librivox.org/`
[6] `http://www.jedit.org/`
[7] `http://notepad-plus-plus.org/`
[8] `http://xml-copy-editor.sourceforge.net/`
[9] `http://www.python.org`
[10] `http://nltk.org/`
[11] `http://calibre-ebook.com/`
[12] `http://code.google.com/p/sigil/`
[13] `http://code.google.com/p/python-epub-builder/`

quedaremos con el lema para crear una versión lematizada de los textos. En las primeras obras estamos utilizando Treetagger (Schmid, 1994), pero se prevee hacer pruebas con Freeling (Carreras et al., 2004) para evaluar si los resultados son más satisfactorios. A medida que aumente el número de lenguas tratadas tendremos que incorporar progresivamente nuevos etiquetadores. No obtante, hay que tener en cuenta que el paso de lematización es opcional.

5 Proceso

5.1 Obtención de las obras

El primer paso que llevaremos a cabo será la obtención de las obras, tanto el original como la traducción. En este paso obtenemos un archivo de texto correspondiente a la obra original y otro archivo correspondiente a la obra traducida. Por el momento simplemente descargamos las obras desde las webs correspondientes. Para agilizar el proceso, en un futuro utilizaremos los *dumps* en XML de Wikisource, que se pueden descargar libremente. Existe un *dump* específico para cada lengua. A partir de estos ficheros podremos obtener las obras originales, y saber a qué lenguas están traducidas. A partir de la información sobre las traducciones podremos acceder a los textos traducidos a partir de los *dumps* correspondientes a cada lengua.

A continuación podemos observar un fragmento de obra en el idioma original:

```
A SCANDAL IN BOHEMIA
I.
To Sherlock Holmes she is always THE woman. I have
seldom heard him mention her under any other name.
In his eyes she eclipses and predominates the whole
of her sex. It was not that he felt any emotion akin
to love for Irene Adler...
```

y el mismo fragmento correspondiente a la traducción:

```
ESCÁNDALO EN BOHEMIA
1.
Ella es siempre, para Sherlock Holmes, la mujer. Rara
 vez le he oído hablar de ella aplicándole otro nombre.
A los ojos de Sherlock Holmes, eclipsa y sobrepasa a
todo su sexo. No es que haya sentido por Irene Adler
nada que se parezca al amor...
```

5.2 Transformación de los textos en ficheros docbook

Docbook[14] (Walsh y Muellner, 1999) es un formato estándar basado en XML que nos permite representar la estructura lógica de un libro. De esta manera se separa totalmente el contenido del formato. La transformación la realizaremos mediante una serie de macros de un editor de texto. Siguiendo el ejemplo anterior, una vez transformados en docbook, los fragmentos tendrían el siguiente aspecto:

```
<chapter>
<title>A SCANDAL IN BOHEMIA</title>
<section>
<title>I.</title>
<para>To Sherlock Holmes she is always THE woman. I
have seldom heard him mention her under any other
name. In his eyes she eclipses and predominates the
whole of her sex. It was not that he felt any emotion
akin to love for Irene Adler...</para>
...
```

y el mismo fragmento correspondiente a la traducción:

```
<chapter>
<title>ESCÁNDALO EN BOHEMIA</title>
<section>
<title>1.</title>
<para>Ella es siempre, para Sherlock Holmes, la mujer.
Rara vez le he oído hablar de ella aplicándole otro
nombre. A los ojos de Sherlock Holmes, eclipsa y
sobrepasa a todo su sexo. No es que haya sentido por
Irene Adler nada que se parezca al amor.</para>
```

Este paso es en realidad opcional, pero preferimos disponer de las obras en este formato ya que existen muchas aplicaciones para transformar documentos docbook en diferentes formatos de salida: html, pdf e incluso epub. El proceso de creación del docbook nos permite revisar a la vez el documento y verificar que la descárga se haya realizado correctamente.

5.3 Segmentación

El siguiente paso consiste en segmentar los textos y transformarlos en un formato de texto que sea adecuado para el alineador automático de textos. Por el momento utilizamos un segmentador genérico del paquete NLTK que nos proporciona buenos resultados. El fichero de salida contiene una oración por línea y adicionalmente, se marcan los párrafos con la marca $< p >$. Los fragmentos anteriores tendrían el siguiente aspecto:

```
A SCANDAL IN BOHEMIA
<p>
I.
<p>
To Sherlock Holmes she is always THE woman.
I have seldom heard him mention her under any other
 name.
In his eyes she eclipses and predominates the whole
of her sex.
It was not that he felt any emotion akin to love for
Irene Adler.
```

[14] http://docbook.org/

y el mismo fragmento correspondiente a la traducción:

```
ESCÁNDALO EN BOHEMIA
1.
<p>
Ella es siempre, para Sherlock Holmes, la mujer.
Rara vez le he oído hablar de ella aplicándole otro
nombre.
A los ojos de Sherlock Holmes, eclipsa y sobrepasa a
todo su sexo.
No es que haya sentido por Irene Adler nada que se
parezca al amor.
```

5.4 Lematización

Los textos obtenidos en el paso anterior ya se podrían alinear sin problemas. Ahora bien, para mejorar los resultados de la alineación es aconsejable lematizar los textos. Para ello se pueden utilizar diversas herramientas, por ejemplo TreeTagger o Freeling. Siguiendo con el ejemplo anterior, obtendríamos el seguiente resultado:

```
A SCANDAL IN BOHEMIA
<p>
 I.
<p>
to Sherlock Holmes she be always the woman .
I have seldom hear him mention her under any other
name .
in his eye she eclipse and predominate the whole
of her sex .
it be not in he feel any emotion akin to love for
Irene Adler .
```

y el mismo fragmento correspondiente a la traducción:

```
ESCÁNDALO EN BOHEMIA
1.
<p>
ella ser siempre para Sherlock Holmes el mujer .
raro vez él haber oír hablar de él aplicar otro
nombre .
a el ojo de Sherlock Holmes eclipsa y sobrepasar
a todo suyo sexo .
no ser que haber sentido por Irene Adler nada que
se parecer al amor .
```

5.5 Alineación

Una vez descargado e instalado Hunalign en nuestro ordenador, ya podremos ejecutar el proceso de alineación mediante una simple orden:

```
hunapertium -utf8 -realign diccionario.dic original.txt
traduccion.txt > alineacion.txt
```

El parámetro *realign* es opcional pero puede mejorar los resultados. Es imprescindible indicar un archivo de diccionario. Los archivos de diccionario tienen la siguiente forma (si es por ejemplo un diccionario para una alineación de inglés a castellano)

```
científico @ scientific
escultórico @ sculptural
estacional @ seasonal
```

Como se puede observar en primer lugar aparecen las palabras en la lengua de llegada. En el caso de no disponer de un diccionario para el par de lenguas de trabajo, se debe indicar igualmente un diccionario que puede ser un fichero vacío.

Los ficheros *original.txt* y *traducción.txt* son los ficheros de texto correspondientes al original y la traducción, que pueden ser simplemente segmentados o bien segmentados y lematizados.

El fichero de alineación tiene el siguiente aspecto:

```
4 0 0
5 1 -0.3
6 1 0
7 2 -0.3
8 2 0
9 2 1.45135
10 3 1.21031
11 4 0.906857
12 5 1.008
```

Se indica la relación entre el número de segmento del fichero original y del fichero traducido y un *score* que indica la calidad de la alineación. Para recuperar un fichero de alineación formado por las oraciones podemos utilizar el *script* llamado *ladder2text.py* que se distribuye con Hunalign.

```
python ladder2text.py alineacion.txt original.txt
traduccion.txt > alineacion_texto.txt
```

Como el orden de los segmentos en la versión lematizada y sin lematizar del original y traducción segmentados son los mismos, si hemos realizado la alineación a partir de los textos lematizados podemos obtener ahora el fichero de alineación con las oraciones sin lematizar indicando en este paso el nombre de los archivos sin lematizar. Al final obtenemos un fichero que relaciona los segmentos originales y traducidos.

```
1.56762 A SCANDAL IN BOHEMIA ESCÁNDALO EN BOHEMIA

1.89878 I 1

1.45135 To Sherlock Holmes she is always THE woman.
Ella es siempre, para Sherlock Holmes, la mujer.

1.21031 I have seldom heard him mention her under any
other name.   Rara vez le he oído hablar de ella aplicándole
otro nombre.

0.906857 In his eyes she eclipses and predominates the
whole of her sex.   A los ojos de Sherlock Holmes, eclipsa
y sobrepasa a todo su sexo.

1.008 It was not that he felt any emotion akin to love
for Irene Adler.   No es que haya sentido por Irene Adler
nada que se parezca al amor.
```

5.6 Creación del html bilingüe

Hemos creado un programa en Python que a partir de los documentos en formato docbook correspondiente al original y la traducción y del fichero de alineación genera un fichero html bilingüe. En este fichero aparecen los segmentos originales y traducidos enlazados. El fichero tiene el siguiente aspecto:

```
<p><a name="s-6"/><a href="#t-2">To Sherlock Holmes she
is always THE woman. </a><a name="s-7"/><a href="#t-3">
I have seldom heard him mention her under any other
name.</a>
<a name="s-8"/><a href="#t-4">In his eyes she eclipses
and predominates the whole of her sex. </a>
...
<p><a name="t-2"/><a href="#s-6">Ella es siempre, para
Sherlock Holmes, la mujer. </a> <a name="t-3"/>
<a href="#s-7">Rara vez le he oído hablar de ella
aplicándole otro nombre.</a><a name="t-4"/>
<a href="#s-8">A los ojos de Sherlock Holmes, eclipsa y
sobrepasa a todo su sexo. </a><a name="t-5"/>
<a href="#s-9">No es que haya sentido por Irene Adler
nada que se parezca al amor. </a></p>
```

En el proceso de alineación es posible que algunos segmentos queden sin alinear. En el caso de no encontrar una alineación válida para un segmento se enlaza con el segmento alineado inmediatamente anterior. De esta manera conseguimos que todos los segmentos estén enlazados entre sí.

En el caso de desear crear un libro en formato mobi, se crea un html bilingüe especial, en el que los enlaces se realizan entre párrafos y mediante una marca formada por un asterisco entre corchetes ([*]). Esto es debido a que tenemos un menor control sobre la salida en mobi y no podemos evitar que los enlaces aparezcan subrayados.

5.7 Creación de los libros electrónicos

Mediante la herramienta Calibre podemos transformar los archivos html bilingües en epub y mobi. También podremos añadir ciertos metadatos y una portada. Todas estas operaciones se pueden realizar desde la interfaz gráfica de usuario, pero puede resultar más cómodo hacerlo en el terminal:

Transformación a epub:

```
ebook-convert sherlock_holmes.html sherlock_holmes.epub
--input-encoding=utf-8 --change-justification=justify
--insert-blank-line --chapter=h1 --chapter-mark=pagebreak
--title="The Adventures of Sherlock Holmes"
--authors="Arthur Conan Doyle" --author-sort=Doyle
--publisher=InLéctor --language=en --cover=portada.jpg
```

Transformación a mobi (utilizaremos el html bilingüe especial alineado por párrafos):

```
ebook-convert sherlock_holmes.html sherlock_holmes.mobi
--input-encoding=utf-8 --change-justification=justify
--insert-blank-line --chapter=h1 --chapter-mark=pagebreak
 --title="The Adventures of Sherlock Holmes"
--authors="Arthur Conan Doyle" --author-sort=Doyle
--publisher=InLéctor --language=en --cover=portada.jpg
```

6 Obras disponibles

Las lenguas de trabajo iniciales de este proyecto serán el inglés, francés y ruso al castellano o catalán dependiendo de la disponibilidad de las traducciones. Las primeras obras publicadas son:

- The Adventures of Sherlock Holmes (Sir Arthur Conan Doyle) (inglés-castellano)
- Sense and Sensibility (Jane Auste) (inglés-castellano)
- Les Trois Mousquetaires (Alexandre Dumas) (francés-castellano)
- Игрок (El jugador) de Fiódor Dostoyevski (ruso-castellano)

En un futuro próximo se trabajará con más lenguas. Para cada obra literaria seleccionada para su publicación se editarán libros bilingües en todas las lenguas cuya traducción esté disponible en Wikisource.

7 Conclusiones y trabajo futuro

En este artículo hemos presentado una metodología automática para la creación de libros electrónicos bilingües, desarrollada en el marco del proyecto InLéctor de la Universitat Oberta de Catalunya. Actualmente el mercado del libro se encuentra inmerso en un cambio de paradigma y el paso de formato papel a formato digital. El avance en este cambio es lento, al menos en nuestro país, ya que en pocos casos la edición digital comporta una mejora substancial para usuario final, ni en precio, ni en funcionalidades. Creemos que dotar al libro electrónico con las funcionalidades previstas en este proyecto puede suponer un impulso para el libro electrónico en nuestro país.

En nuestro proyecto tratamos únicamente con obras en dominio público, tanto en lo que hace referencia al original como a la traducción. Las editoriales que dispongan de los derechos de autor y de traducción de una obra se pueden beneficiar de nuestras propuestas y publicar las obras de su catálogo en este formato. El libro digital se asemejaría a una película en DVD, al hacer posible que el usuario pueda escoger la lengua

y los subtítulos y adaptar la visualización a sus preferencias.

Actualmente el proyecto no cuenta con financiación específica por lo que el avance es lento. Nuestra intención es obtener financiación externa para mejorar las herramientas de creación de libros paralelos. El equipo investigador está abierto a colaboraciones externas ya sea en la mejora de las herramientas como en la creación de nuevas obras.

Los siguientes pasos que llevaremos a cabo serán en las siguientes direcciones:

- Automatizar la obtención de las obras originales y traducidas mediante los *dump xml* de Wikisource
- Mejorar el algoritmo de creación de libros electrónicos
- Añadir enlaces al audio correspondiente a la lectura humana del original

La nuevas especificaciones del formato epub3 (Garrish, 2012) facilitarán enormemente la integración de todas estas funcionalidades.

Una vez desarrolladas las funcionalidades previstas y creadas un número suficiente de obras se pretende fomentar el uso de estas obras en aulas de enseñanza de idiomas. En las últimas etapas del proyecto llevarán a cabo experimentos psicolingüísticos para evaluar las mejoras en la adquisición de léxico y estructuras gramaticales por parte de los estudiantes de idiomas.

Bibliografía

Bond, Francis y Kyonghee Paik. 2012. A survey of wordnets and their licenses. En *Proceedings of the 6th Global WordNet Conference (GWC 2012)*, páginas 64–71, Matsue, Japan.

Carreras, X., I. Chao, L. Padró, y M. Padró. 2004. Freeling: An open-source suite of language analyzers. En *Proceedings of the 4th LREC*, volumen 4.

Forcada, M. L, F. M Tyers, y G. Ramírez. 2009. The apertium machine translation platform: five years on. En *Proceedings of the First International Workshop on Free/Open-Source Rule-Based Machine Translation*, página 3–10.

Garrish, Matt. 2012. *Accessible EPUB 3.* O'Reilly Media.

Loper, E. y S. Bird. 2002. NLTK: the natural language toolkit. En *Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics-Volume 1*, página 63–70.

Oliver, A., M. Coll-Florit, y S. Climent. 2012. Inléctor: Sistema de lectura bilingüe interactiva. *Procesamiento del Lenguaje Natural*, 49:279–286, September.

Schmid, H. 1994. Probabilistic part-of-speech tagging using decision trees. En *Proceedings of international conference on new methods in language processing*, volumen 12, página 44–49.

Varga, D., P. Halácsy, A. Kornai, V. Nagy, L. Németh, y V. Trón. 2007. Parallel corpora for medium density languages. En *Proceedings of the RANLP 2005*, páginas 590–596.

Walsh, N. y L. Muellner. 1999. *DocBook: The definitive guide*, volumen 1. O'Reilly Media.

ECPC: el discurso parlamentario europeo desde la perspectiva de los estudios traductológicos de corpus*

José Manuel Martínez Martínez
Universität des Saarlandes
j.martinez@mx.uni-saarland.de

Iris Serrat Roozen
Universidad de Valencia
iris.serrat@uv.es

Resumen

Este artículo presenta la labor investigadora del grupo ECPC, que ha diseñado y creado un Archivo de discursos parlamentarios europeos con el fin de estudiar dicho género y la hipotética influencia de la traducción en la construcción de la identidad europea. La investigación se ha restringido al Parlamento Europeo (mediante la construcción de un corpus paralelo —EN y ES— con las versiones en inglés y español) y a dos parlamentos nacionales, la House of Commons británica (HC) y el Congreso de los Diputados español CD), que constituyen sendos corpus comparables. El Archivo contiene los discursos recogidos en las actas de las sesiones plenarias celebradas a lo largo de la VI legislatura del Parlamento Europeo (2004-2009) en cada una de las cámaras anteriormente mencionadas.

Palabras clave

Estudios Traductológicos de Corpus, metodología, discurso político, debate parlamentario, Parlamento Europeo, House of Commons, Congreso de los Diputados, Análisis Crítico del Discurso, corpus comparable, corpus paralelo

Abstract

This paper presents the main outcome of the ECPC research group: an archive of European parliamentary speeches created to study this genre and the hypothetical influence of translation in the construction of European identity. The archive is made up of, on the one hand, a parallel corpus containing the English and Spanish versions of the European Parliament proceedings, and on the other hand, two comparable corpora —one containing the proceedings of the House of Commons for English and the proceedings of the Congreso de los Diputados for Spanish. The archive contains the speeches delivered in the plenary sittings held during the 6th term of the European Parliament (2004-2009) before each of the above mentioned Houses.

Keywords

Corpus-based Translation Studies, methodology, political discourse, parliamentary debate, European Parliament, House of Commons, Congreso de los Diputados, Critical Discourse Analysis, comparable corpus, parallel corpus

1 Grupo ECPC

El grupo de investigación ECPC[1] (European Parliamentary Comparable and Parallel Corpora) inició oficialmente su andadura en el año 2005, cuando recibió financiación por parte del Ministerio de Educación y Ciencia para el proyecto Corpus comparables y paralelos de discursos parlamentarios europeos, con referencia HUM 2005-03756/FILO.

En la actualidad, el grupo está consolidando su actividad investigadora gracias a la financiación del Ministerio de Ciencia e Innovación bajo el nuevo título Ampliación y profundización de ECPC y de ConcECPC 1.0: avances teórico-descriptivos e innovaciones tecnológicas, con referencia FFI2008-01610/FILO.

Este proyecto se enmarca dentro de los Estudios Traductológicos de Corpus, de manera que la investigación en el seno del grupo se realiza a partir de corpus comparables y paralelos los cuales están compuestos por discursos del Parlamento Europeo (EN para la versión en inglés y ES para la versión en español), así como de los parlamentos nacionales de España (Congreso de los Diputados – CD) y del Reino Unido (Cámara de los Comunes/House of Commons – HC).

*El Archivo ECPC se ha creado en el marco del proyecto Ampliación y Profundización de ECPC y de ConcECPC 1.0 (FFI2008-01610/FILO) financiado por el Ministerio de Ciencia e Innovación durante los años 2009-2011.

[1]Web con información acerca de ECPC: http://www.ecpc.uji.es

2 Antecedentes

2.1 Lingüística de corpus (LC)

El proyecto ECPC es heredero del trabajo que iniciaran en el ámbito de la Lingüística de Corpus expertos como Henry Kučera y W. Nelson Francis (Brown University, Providence, Rhode Island), quienes compilaron por primera vez un corpus electrónico, el Brown Corpus, formado por un millón de palabras de inglés estadounidense. También su homólogo británico, el Lancaster-Oslo/Bergen Corpus (LOB), creado por Geoffrey Leech (Lancaster University), Stig Johansson (Universitetet i Oslo) y Knut Hofland (Universitetet i Bergen) es un referente ineludible.

2.2 Estudios Traductológicos de Corpus (Corpus-based Translation Studies, CTS)

ECPC se nutre asimismo de las aportaciones realizadas por parte de los Estudios Traductológicos de Corpus, como es el Translational English Corpus (TEC)[2], elaborado bajo la dirección de Mona Baker (Baker, 2004), que ha servido de objeto de estudio para otras investigadoras como Kenny (2001), Laviosa (2002) y, posteriormente, Saldanha (2004) y Winters (2004). TEC puede consultarse a través de la interfaz web desarrollada por Luz (2000). Sin embargo, este corpus monolingüe sólo contiene traducciones. ECPC, da un paso más y ofrece también los textos originales (lo que conforma el corpus paralelo EN-ES) y textos producidos por hablantes nativos (lo que da lugar a sendos corpus comparables en español CD, e inglés HC). Precisamente, esta arquitectura que caracteriza al Archivo ECPC, un corpus paralelo más corpus comparables, se ha tomado del English-Nowegian Parallel Corpus (ENPC), la aportación del tándem Johansson y Oksefjell (2000).

En una línea muy similar pero con el fin de estudiar la interpretación en el Parlamento Europeo se enmarca el Eropean Parliament Interpreting Corpus (EPIC)[3] (Sandrelli, Bendazzoli, y Russo, 2010). Su corpus, de un tamaño más modesto que ECPC, recoge transcripciones de los discursos originales y sus interpretaciones en inglés, español e italiano en las que se ha anotado información relacionada con los oradores, rasgos propios de la interpretación y se han etiquetado los textos morfosintácticamente. Además es posible consultarlo utilizando una interfaz web creada al efecto.

2.3 Procesamiento del Lenguaje Natural (PLN)

ECPC también se ha inspirado en propuestas procedentes del PLN como el corpus Europarl (Koehn, 2005). Este Archivo, compuesto por las actas del Parlamento Europeo en 11 idiomas alineadas al nivel de la frase, fue creado con el fin de entrenar sistemas estadísticos de traducción automática. Se ha seguido una metodología similar en cuanto a la obtención del material en bruto y su preparación para el alineado. No obstante, dado que los objetivos investigadores son diferentes, ha sido preciso adaptar esa propuesta y ampliarla tal y como se describe en el apartado 5 de metodología, poniendo un acento especial en la obtención y anotación de información metatextual acerca de los oradores no contenida en los textos y en la revisión del etiquetado y el alineado del corpus paralelo.

Tiedemann y Nygaard (2004) aprovechan el corpus recopilado por Koehn y lo incluyen en su colección de corpus libres, OPUS[4]. Los dos investigadores procesaron este material y lo pusieron a disposición de la comunidad científica, que puede descargarlo en diferentes formatos o consultarlo mediante una interfaz web que utiliza el Corpus Workbench (CWB). Otros grupos de investigación también han hecho accesible este corpus (o versiones más recientes) mediante interfaces web[5][6].

3 Objetivos

3.1 Objetivo principal

El objetivo principal del grupo ECPC es conocer y profundizar en el estudio del discurso parlamentario como género textual con especial atención a la influencia de la traducción en dicho género.

[2]Web con información acerca de TEC y acceso a la herramienta para su consulta: `http://www.monabaker.com/tsresources/TranslationalEnglishCorpus.htm`

[3]Se puede obtener más información y consultar el corpus en `http://sslmitdev-online.sslmit.unibo.it/corpora/corporaproject.php?path=E.P.I.C.`

[4]Interfaz web para la versión 3 de Europarl basada en el CWB `http://opus.lingfil.uu.se/bin/opuscqp.pl?corpus=Europarl3`

[5]La versión 5 del mismo corpus accesible gracias al proyecto Per-Fide en `http://per-fide.di.uminho.pt/query/`

[6]Los desarrolladores del CWB también ofrecen una interfaz web para la versión 3 del Europarl en `http://linglit193.linglit.tu-darmstadt.de/CQP/Europarl/frames-cqp.html`

3.2 Objetivos secundarios

- Crear un Archivo en formato electrónico compuesto por diversos corpus que permita la combinación y comparación de los mismos.
- Desarrollar parámetros para realizar estudios contrastivos a partir de los corpus comparables y paralelos que conforman dicho Archivo, entre los que se destacan los siguientes objetivos investigadores:
  - Examinar el grado de similitud y/o diferencia entre los discursos emitidos en el Parlamento Europeo y aquellos emitidos en parlamentos nacionales de diferentes Estados Miembros (el Congreso de los Diputados español y la Cámara de los Comunes británica).
  - Establecer una comparación de la representación de la identidad europea entre los distintos parlamentos objeto de estudio.
  - Realizar un estudio del discurso y de la ideología en los diferentes parlamentos.
- Difundir tanto el conocimiento derivado de la realización de los estudios contrastivos anteriores como los resultados del proyecto.
- Desarrollar herramientas de análisis accesibles vía web, con el fin de permitir la consulta del Archivo y la replicación de los estudios realizados. Dichas herramientas deben facilitar la generación en línea de concordancias monolingües y paralelas y la obtención de información estadística relevante para describir y comparar los fenómenos estudiados.
- Elaborar recursos de ayuda a la traducción como memorias de traducción, glosarios, etc.
- Originar un recurso de referencia para las actividades relacionadas con el Procesamiento del Lenguaje Natural (como la traducción automática o la extracción terminológica, entre otras).
- Diseñar propuestas didácticas en torno a la traducción del discurso parlamentario como género textual dirigidas tanto a estudiantes de traducción como a traductores profesionales.

4 Descripción del Archivo

El Archivo de discursos parlamentarios ECPC está compuesto por diferentes corpus que han sido recopilados en formato electrónico. Estos corpus son:

- Discursos procedentes de la Cámara Baja británica (House of Commons, HC)
- Discursos procedentes de la Cámara Baja española (Congreso de los Diputados, CD)
- Discursos en español procedentes del Parlamento Europeo (ES)
- Discursos en inglés procedentes del Parlamento Europeo (EN)

La muestra descargada contiene los discursos emitidos a lo largo del periodo correspondiente a la VI legislatura del Parlamento Europeo (20 de julio de 2004 al 30 de julio de 2009).

Los textos procedentes del Parlamento Europeo reúnen una serie de características que los hacen únicos y que conviene señalar. En primer lugar, la lengua en la que se expresan los oradores a menudo no es su lengua materna, por lo que podemos encontrarnos ante textos producidos por hablantes no nativos. En segundo lugar, las actas no son transcripciones literales de lo dicho por el orador, ni de las interpretaciones. Toda intervención pasa por un proceso de edición que consiste en:

1. La transcripción y corrección del texto oral en la lengua original a partir de la grabación de la sesión siguiendo una serie de normas[7] (el resultado se conoce como versión "arcoíris").
2. La traducción de cada intervención al inglés y, posteriormente, la traducción desde la versión inglesa al resto de lenguas oficiales.

Los rasgos anteriormente descritos dificultan la distinción entre textos influidos por la traducción de aquellos que no lo han sido. Aunque en la mayoría de casos se conoce la lengua original en la que el orador dio su discurso no podemos tener la certeza de si el orador habló en su lengua materna o en una segunda lengua. Sí que podemos saber, sin embargo, si una intervención se ha visto afectada en mayor o menor medida por los procesos de traducción que se realizan en el seno de la Dirección General de Traducción del Parlamento Europeo desde la emisión del discurso oral hasta su publicación en la página web.

Es decir, si en la versión española de las actas encontramos una intervención en español, podemos afirmar que la versión publicada no ha sido mediada por un proceso de traducción (en verde en el cuadro 1). Si la lengua original fue el

[7] Véase `http://www.europarl.europa.eu/transl_es/plataforma/pagina/guia/cre_normas.htm`

inglés, podemos afirmar que se trata de una traducción directa del inglés al español (en amarillo). Sin embargo, para las demás intervenciones pronunciadas en una lengua distinta al español o al inglés, sólo podemos decir que se trata de una intervención que ha sido mediada por un proceso de traducción indirecta en el que el inglés ha sido la lengua pivote (en rojo).

Audio	Arcoiris	EN	ES	IT	DE
EN	EN	EN	ES	IT	DE
ES	ES	EN	ES	IT	DE
IT	IT	EN	ES	IT	DE
DE	DE	EN	ES	IT	DE

Cuadro 1: Grado de mediación interlingüística en las actas del Parlamento Europeo publicadas en Internet

En cuanto al tamaño, para el Archivo, que comprende las actas desde 2004 a 2009 de cada uno de los Parlamentos estudiados, el número de *tokens* por cada corpus puede apreciarse en el cuadro 2:

Corpus	# Tokens
EN	21 737 797
ES	22 685 242
HC	47 712 000
CD	23 734 230

Cuadro 2: Tamaño del corpus ECPC para el periodo 2004-2009

5 Metodología

El grupo ECPC utiliza la metodología de los Estudios Traductológicos de Corpus para estudiar el discurso parlamentario como género textual. Al aplicar dicha metodología de trabajo hemos pretendido superar el tradicional dilema: primar o bien la calidad o bien el tamaño del Archivo. Para obtener un corpus de un tamaño suficiente y que respondiese a los fines perseguidos por el proyecto hubo que ir más allá de los métodos de etiquetado manual tradicionales en nuestro ámbito. A continuación describimos someramente el proceso seguido en este proyecto para obtener nuestro Archivo que se compone de 6 fases: 1) recopilación; 2) almacenamiento; 3) transformación en XML; 4) enriquecimiento de los textos con metadatos sobre los oradores; 5) control de calidad y; 6) alineado de los corpus paralelos.

5.1 Recopilación del Archivo

Estos discursos se han descargado de los diarios de sesiones en formato electrónico (documentos en HTML) accesibles en las respectivas páginas web de cada uno de los diferentes parlamentos. Se eligió el formato HTML frente al PDF pues el primero se puede manipular más fácilmente y el texto está más limpio que en el caso del segundo lo cual facilitó su procesamiento posterior. Por otra parte, para automatizar esta fase se utilizó un *web crawler* al que se le suministró un listado con todas las direcciones que debía descargar. Finalmente se obtuvo un único documento para cada sesión plenaria que contenía todas las intervenciones realizadas durante ese día.

5.2 Almacenamiento del material

Una vez obtenidos los textos que configuran nuestro Archivo, se hizo necesario utilizar un sistema que nos permitiera almacenar toda esta información y que además, dada la naturaleza del grupo con miembros radicados en distintas universidades europeas, estuviese disponible y al alcance de todos ellos. En concreto, buscamos una solución que nos permitiese: 1) acceder al material desde cualquier lugar; 2) poder compartirlo con el resto de miembros; 3) contar con un historial de versiones y; 4) un sistema de control de cambios.

La tecnología elegida para cubrir estas necesidades fue un repositorio *Subversion*, de gran éxito en el ámbito del desarrollo de software.

De este modo todos los investigadores tenían acceso a una copia central alojada en un servidor corporativo de la Universitat Jaume I a partir de la cual podían generar una copia local sobre la que realizar los cambios y posteriormente compartirlos con el resto de miembros. En todo momento, el sistema permitía registrar un historial con todas las modificaciones y gestionar los cambios pudiendo comprobar en qué habían consistido los mismos, deshacerlos, etc.

5.3 Transformación en XML

En este fase del proyecto el objetivo consistió en estructurar, limpiar y anotar automáticamente información metatextual contenida en los mismos textos recopilados. Para ello se identificaron patrones formales en el código fuente HTML de las páginas descargadas previamente, se escribieron expresiones regulares para anotar la información deseada en XML mediante operaciones de búsqueda y reemplazo y, finalmente, se encadenaron en *scripts* de Perl para procesar los documentos de cada subcorpus por lotes. Se eligió el formato XML frente al convencional texto plano para poder describir con precisión la estructura de los documentos, anotar información

sobre los participantes en la situación comunicativa y facilitar el procesamiento del corpus en fases posteriores.

La transformación en XML de los textos y el etiquetado de la información deseada no fueron tareas sencillas porque a lo largo del tiempo el formato de las actas ha ido sufriendo leves modificaciones, por un lado y, por otro, las marcas HTML no siempre seguían de forma sistemática los patrones generales que se habían identificado inicialmente, introduciendo ruido en el XML resultante. Estos hechos obligaron a pilotar y corregir los *scripts* para mejorar el rendimiento de los mismos.

Finalmente, en esta primera fase se pudo extraer la información siguiente:

- En cuanto a los textos: orden del día, encabezados para cada punto del orden del día, intervenciones, la lengua (o lenguas) de cada intervención, el modo en que cada intervención fue presentada (oral o por escrito), comentarios acerca de los procedimientos de las cámaras o acciones y/o reacciones de los oradores.
- En cuanto a los oradores: nombre, grupo parlamentario y cargo.

5.4 Enriquecimiento del etiquetado

A continuación se procedió a enriquecer el etiquetado obtenido en la fase anterior con más información acerca de los oradores que habían participado en cada debate. Dado que ya se habían anotado los nombres de los oradores responsables de cada intervención se pudieron añadir más datos acerca de los mismos tales como: tratamiento, afiliación política, sexo, fecha de nacimiento, lugar de nacimiento y país de procedencia. Estos datos se suministraron a partir de una base de datos creada al efecto con información extraída o bien de las webs oficiales de las distintas cámaras (CD y EN/ES) o bien proporcionada por los servicios de documentación (HC).

Para obtener esa información adicional, en el caso del CD y el EN/ES, se volvió a emplear un *web crawler* para descargar la página web en HTML que contenía la información personal oficial de cada diputado tal y como las proporcionaba la cámara correspondiente. Se extrajo la información necesaria mediante un *script* de Perl basado en búsquedas de patrones y se estructuró de forma tabulada. Para hacer efectiva la incorporación de la nueva información otro *script* de Perl leía las etiquetas que señalaban los nombres de los oradores en las actas en XML, buscaba en la base de datos dicho nombre y si lo encontraba recuperaba el resto de información y la incorporaba al texto en forma de etiquetas XML. En el caso del HC se modeló la información proporcionada en forma de hojas de cálculo de modo que pudiese ser utilizada por el mismo *script* al que nos acabamos de referir.

En la eventualidad de que un orador no apareciese en la base de datos porque no pertenecía al organismo objeto de estudio (tales como miembros del Gobierno, expertos, Comisarios, miembros del Consejo, etc.) se completó su información con los datos proporcionados en las páginas web oficiales de la institución a la que pertenecían y cuando no fue posible se acudió a fuentes secundarias como la Wikipedia.

5.5 Control de calidad

Acorde con la idea de obtener un corpus de cientos de millones de *tokens* sin sacrificar la calidad, se impuso un método de revisión para comprobar que el etiquetado realizado era lo suficientemente preciso y ayudaba a extraer toda la información contenida en los textos.

De nuevo el marcado en XML de los textos volvió a ser una ventaja pues permitió fácilmente comprobar si los documentos estaban bien formados y si seguían las especificaciones recogidas en la DTD que describía el juego de etiquetas empleado en nuestro corpus.

Para el CD y el HC se realizó un único control de calidad en este punto. Sin embargo, para las versiones inglesa y española del Parlamento Europeo, se realizaron dos. La primera antes del paso descrito en el apartado 5.4, consistente en comprobar que el XML estaba bien formado y era válido, junto con la comparación de la estructura del etiquetado de cada bitexto. Esta comprobación extra se realizó para detectar cualquier tipo de error generado en la fase 5.4 que pudiese afectar al alineado, como intervenciones que habían quedado sin detectar, frases de texto etiquetadas como comentarios o notas, etc. La segunda revisión se realizó tras enriquecer el etiquetado limitándose a comprobar la corrección del XML en términos de validez y forma.

5.6 Alineado de los corpus paralelos

Para el alineado de las versiones española e inglesa de los debates del Parlamento Europeo se empleó como gestor y editor del alineado InterText server[8], creado por Pavel Vondřička. Esta

[8] InterText server `http://wanthalf.saga.cz/intertext#ITserver`

herramienta, desarrollada en el marco del proyecto InterCorp, está concebida precisamente para gestionar el alineado de corpus paralelos multilingües, es muy flexible en cuanto al formato de entrada de los textos, siempre que estén anotados en XML, y soporta la codificación de caracteres Unicode. Esta herramienta es un sitio web dinámico basado en PHP y MySQL. Esta arquitectura permite que tanto los administradores y coordinadores del flujo de trabajo como los editores encargados de la revisión del alineado puedan trabajar en línea desde distintos lugares utilizando una GUI intuitiva. Además, la herramienta cuenta con una CLI muy útil que permite realizar las principales tareas administrativas por lotes como la importación de las dos versiones de cada bitexto, el alineado y la exportación del resultado.

InterText deja el alineado automático en manos de dos potentes alineadores, HunAlign (Varga et al., 2005) y TCA2 (Hofland y Johansson, 1998), que se pueden integrar en el sistema. El primero emplea un algoritmo estadístico que devuelve con relativa rapidez un alineado aceptable. El segundo se sirve de un enfoque más sofisticado basado en un conjunto de algoritmos que junto con un diccionario de términos sopesa distintas opciones de las que elije aquella que ha obtenido una mejor puntuación. Aunque TCA2 necesita mucho más tiempo para alinear un mismo texto, se optó por este alineador pues el resultado final es más fiable, de cara a su revisión manual. En comparación, HunAlign tiende a crear relaciones de alineado 1:1 erróneas que pueden resultar indetectables cuando se sigue el proceso de revisión que explicamos a continuación.

Para poder alinear los textos en primer lugar se dividió, mediante un *script* de Perl y de forma automática, cada intervención en párrafos y frases, siendo esta última división la unidad mínima de alineado. Posteriormente se importó en InterText todo el corpus utilizando el comando *import* de la CLI y se alineó automáticamente el corpus paralelo con el alineador TCA2. El uso de este alineador en concreto permitió que la labor de revisión y edición del alineado se limitara a aquellos segmentos que no fueran relaciones 1:1 (una unidad del texto original alineada con una unidad del texto traducido).

Para los textos correspondientes a los años 2004, 2006, 2007, 2008 y 2009 se alinearon un total de 238 textos del corpus EN con los correspondientes a la versión ES. El alineado automático produjo un total de 589 665 segmentos, de los cuales un 97,66 % de los casos consistió en relaciones 1:1 y el 2,33 % restante correspondió a otro tipo de relaciones. Sólo se revisaron manualmente por medio de la GUI estas últimas, puesto que InterText permite encontrar las relaciones que no son 1:1 directamente.

En la revisión se comprobó: 1) que la división en frases fuese correcta; 2) que la propuesta de alineado fuese correcta en cuanto al contenido.

En el primer nivel de revisión, si la división era incorrecta debido a que las reglas del *script* se habían topado con un caso no previsto se procedió a separar o fusionar las frases afectadas. Si el problema se había producido por la falta de signos de puntuación que delimitasen el final de la frase se comprobó contrastando con el HTML original si la puntuación se había perdido en alguno de los procesos de transformación anteriores o bien se trataba de una errata del original. Si el error se debía a un fallo introducido durante el procesado de los textos se corrigió tanto aquel elemento que había generado el error como la división. Si el error se debía, por el contrario, a una errata la división se corrigió sin añadir la puntuación que en teoría faltaba en el original.

En el segundo nivel de revisión, se detectaron los siguientes tipos de relaciones:

ES≥1:EN=0 una frase o más en español por ninguna en inglés.

ES=0:EN≥1 ninguna frase en español por una frase o más en inglés.

ES=1:EN>1 una frase en español por varias frases en inglés.

ES>1:EN=1 varias frases en español por una frase en inglés.

ES>1:EN>1 más de una frase en español para más de una frase en inglés.

Rel. 1≠1	no rev.	rev.	no rev.	rev.
ES≥1:EN=0	620	38	4,50 %	0,35 %
ES=0:EN≥1	1126	182	8,17 %	1,69 %
ES=1:EN>1	7517	6808	54,57 %	63,37 %
ES>1:EN=1	4510	3637	32,74 %	33,85 %
ES>1:EN>1	0	77	0 %	0,71 %

Cuadro 3: Relaciones no 1:1, diferencias entre el resultado del alineado automático con TCA2 (no rev.) y la revisión manual posterior (rev.)

Se verificó que para ninguno de los 5 casos la relación no 1:1 se debiese a una propuesta errónea de TCA2. Si se detectaba algún fallo en este sentido (una o más frases asociadas a un segmento contiguo que en realidad pertenecían al segmento objeto de revisión) se corrigió utilizando el editor de InterText. Para los casos 1 y 2 se comparó además con la versión HTML original con el

fin de descartar que la omisión de información en una de las versiones se debiese al procesado previo de los textos. Los fenómenos del tipo 5 recogen relaciones complicadas (Frankenberg-Garcia, Santos, y Silva, 2006, pp. 8-10) como alineados con frases enteras y fracciones y reordenaciones. En nuestro caso, dada la baja incidencia de esta clase de segmentos no hemos realizado intervenciones más allá de lo expuesto con anterioridad. Además, puesto que InterText registra los cambios realizados, no se anotaron aquellos segmentos que sufrieron modificaciones durante el proceso de revisión.

Tras la revisión se obtuvo un total de 589 445 segmentos alineados, de los cuales un 98,14 % consistió en relaciones 1:1 y el 1,82 % restante correspondió a otro tipo de relaciones. Dicha revisión arroja un total de 2809 diferencias en cuanto a la segmentación en frases y las relaciones de alineado propiamente dichas.

Por último, se exportaron los textos alineados en tres formatos distintos:

corresp: se obtiene un documento por idioma, con los identificadores de los elementos alineables actualizados (en nuestro caso las frases) donde la información sobre el alineado se codifica mediante un atributo llamado "corresp"que indica el identificador de las unidades equivalentes en la otra lengua. Este formato es usado como input para Glossa.

segs: se obtiene un documento por idioma, con los identificadores de las frases actualizados donde la información sobre el alineado se codifica en el mismo texto utilizando unos elementos llamados "seg"para delimitar las areas de texto equivalentes en cada versión. Este formato es el que puede utilizar ParaConc como input.

TEI alignment format: se obtiene un documento por idioma, con los identificadores de las frases actualizados, pero los detalles sobre el alineado se almacenan en un tercer documento en formato XML. Este formato permite volver a importar el alineado en InterText y sigue la recomendación recogida en TEI para codificar este tipo de información.

5.7 Desarrollo de software de consulta

En la actualidad Luz (Calzada Pérez y Luz, 2006) se encarga de desarrollar un conjunto de herramientas que posibilitarán la consulta del corpus vía web y que incorpora distintas características como la generación de concordancias monolingües, la selección de distintos subcorpus atendiendo a las variables codificadas como información metatextual y la representación visual de distintos tipos de información lingüística.

Al mismo tiempo Anders Nøklestad trabaja en la adaptación de Glossa (Nygaard et al., 2008) con el fin de facilitar la generación de concordancias paralelas con caracteríscas similares a la herramienta de Luz. Glossa es una interfaz web que emplea el Open Corpus WorkBench (Evert y Hardie, 2011) como motor de búsqueda y gestor del corpus.

6 Aplicaciones

6.1 Aplicación en la investigación

Entre las potenciales aplicaciones investigadoras de este corpus cabe destacar la posibilidad de profundizar en el conocimiento sobre el género de los discursos parlamentarios (tanto del Parlamento Europeo como de otros parlamentos de Estados Miembros) en una línea similar a los trabajos de Partington (2003) y Guerini, Strapparava, y Stock (2008); y además examinar la influencia de la traducción (Calzada Pérez, 2007). Sin embargo, este material gracias el etiquetado metatextual acerca de los oradores puede ser de interés no sólo para estudiosos de la traducción y la lingüística sino también de la sociolingüística, sociología e incluso de los estudios de género.

6.2 Aplicación en la didáctica

Los potenciales beneficiarios de esta herramienta no se limitan al mundo científico pues los profesionales del ámbito de la traducción, los centros de formación de traductores y los aprendices de esta disciplina también podrán consultar el material utilizando una interfaz web de consulta similar a la del BYU-BNC de Mark Davies[9] o Glossa (Nygaard et al., 2008). Este enfoque ya ha sido explotado con éxito en el ámbito de la enseñanza de lenguas (Moreno Jaén, Serrano, y Calzada Pérez, 2010) y de la traducción (Zanettin, Bernardini, y Stewart, 2003; Beeby, Rodríguez Inés, y Sánchez-Gijón, 2009).

7 Perspectivas de futuro

Algunas de las tareas que el grupo ECPC está realizando en la actualidad o que tiene previstas realizar en el futuro son:

[9]BYU-BNC: The British National Corpus `http://corpus.byu.edu/bnc`

1. Clasificación temática de las intervenciones para poder agruparlas en subcorpus "especializados" utilizando el JRC EuroVoc Indexer (Pouliquen, Steinberger, y Ignat, 2003).
2. Ampliación del Archivo con la versión en alemán de los discursos del PE (DE) y su homólogo nacional, el Bundestag alemán (DB) para el mismo periodo de tiempo.
3. Etiquetado morfosintáctico de todos los corpus que componen el Archivo con TreeTagger[10].

Cabe reseñar la creación de dos corpus derivados de la experiencia acumulada en el seno del grupo ECPC: EMPAC y TraDiCorp.

El EMPAC (EuroparlTV Multimedia Parallel Corpus) es un corpus multilingüe de subtítulos de las noticias emitidas en el canal EuroparlTV del Parlamento Europeo. Actualmente el corpus presenta una versión etiquetada del año 2010 en inglés y en español.

El TraDiCorp (Translation Difficulties Corpus) es un corpus paralelo inglés-español de múltiples traducciones de textos de las actas del Parlmento Europeo realizadas por estudiantes de grado y máster de traducción, con problemas de traducción anotados por los mismos estudiantes en el texto original.

Bibliografía

Baker, Mona. 2004. A corpus-based view of similarity and difference in translation. *International Journal of Corpus Linguistics*, 9(2):167–193.

Beeby, Allison, Patricia Rodríguez Inés, y Pilar Sánchez-Gijón. 2009. *Corpus use and translating: corpus use for learning to translate and learning corpus use to translate*, volumen v. 82 de *Benjamins translation library*. John Benjamins, Amsterdam.

Calzada Pérez, María. 2007. *Transitivity in translating: the interdependence of texture and context*. Peter Lang, Bern/Berlin/Bruxelles/Frankfurt am Main/New York/Oxford/Wien.

Calzada Pérez, María y Saturnino Luz. 2006. ECPC: Technology as a tool to study the (linguistic) functioning of national and transnational European parliaments. *Journal of Technology, Knowledge and Society*, 5(2):53–62.

Evert, Stefan y Andrew Hardie. 2011. Twenty-first century Corpus Workbench : Updating a query architecture for the new millennium. En *Proceedings of the Corpus Linguistics 2011 conference*, Birmingham, UK.

Frankenberg-Garcia, Ana, Diana Santos, y Rosario Silva. 2006. COMPARA: Sentence alignment revision and markup. Informe técnico, Linguateca.

Guerini, M, C Strapparava, y O Stock. 2008. CORPS: A corpus of tagged political speeches for persuasive communication processing. *Journal of Information Technology & Politics*, 5(1):19–32.

Hofland, Knut y Stig Johansson. 1998. The Translation Corpus Aligner: A program for automatic alignment of parallel texts. En Stig Johansson y Signe Oksefjell, editores, *Corpora and Cross-linguistic research: Theory, Method and Case Studies*, volumen 24. Rodopi, Amsterdam; New York, páginas 87–100.

Johansson, Stig y Signe Oksefjell. 2000. The English-Norwegian Parallel Corpus: Current Work And New Directions. En S Botley McEnery A., y A Wilson, editores, *Multilingual corpora in teaching and research*. Rodopi, Amsterdam; Atlanta, páginas 134–147.

Kenny, Dorothy. 2001. *Lexis and creativity in translation: a corpus-based study*. St. Jerome, Manchester.

Koehn, Philipp. 2005. Europarl: A parallel corpus for statistical machine translation. En *Proceedings of the Tenth Machine Translation Summit (MT Summit X)*, volumen 5, Phuket, Tailandia, Septiembre. Asia-Pacific Association for Machine Translation and Thai Computational Linguistics Laboratory.

Laviosa, Sara. 2002. *Corpus-based translation studies: theory, findings, applications*. Rodopi, Amsterdam; New York.

Luz, Saturnino. 2000. A software toolkit for sharing and accessing corpora over the Internet. En M Gavrilidou G Carayannis S Markantonatou S Piperidis, y G Stainhauer, editores, *Proceedings of the Second International Conference on Language Resources and Evaluation, LREC 2000*, Athenas, Greece, Mayo. European Language Resources Association (ELRA).

Moreno Jaén, María, Fernando Serrano, y María Calzada Pérez. 2010. *Exploring new paths in language pedagogy: lexis and corpus-based language teaching*. Equinox, London.

[10] TreeTagger `http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger`

Nygaard, L, J Priestley, A Nø klestad, y J B Johannessen. 2008. Glossa: A multilingual, multimodal, configurable user interface. En *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC 2008)*. European Language Resources Association (ELRA), Marrakesh, Morocco, Mayo.

Partington, Alan. 2003. *The Linguistics of Political Argument: the Spin-Doctor and the Wolf-Pack at the White House*. Routledge, London.

Pouliquen, Bruno, Ralf Steinberger, y Camelia Ignat. 2003. Automatic annotation of multilingual text collections with a conceptual thesaurus. En *Workshop Ontologies and Information Extraction at the Summer School The Semantic Web and Language Technology - Its Potential and Practicalities (EUROLAN 2003)*, Bucarest, Romania.

Saldanha, G. 2004. The Translator's Presence in the Text: A Corpus-based Exploration. En *1st IATIS conference: Translation and the Construction of Identity*, Seul, South Korea, Agosto. International Association for Translation and Interecultural Studies (IATIS).

Sandrelli, A, C Bendazzoli, y M Russo. 2010. European Parliament Interpreting Corpus (EPIC): Methodological Issues and Preliminary Results on Lexical Patterns in Simultaneous Interpreting. *International Journal of Translation Studies*, 22(1-2):167–206.

Tiedemann, J y L Nygaard. 2004. The OPUS corpus–parallel and free. En M T Lino M F Xavier F Ferreira R Costa, y R Silva, editores, *In Proceeding of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, Lisbon, Portugal, Mayo. European Language Resources Association (ELRA).

Varga, D, P Halácsy, A Kornai, V Nagy, L Németh, y V Trón. 2005. Parallel corpora for medium density languages. En *Proceedings of the RANLP 2005*, Borovets, Bulgaria, Septiembre.

Winters, Marion. 2004. F. Scott Fitzgerald's Die Schönen und Verdammten – A Corpus based Study of Translators' Style: Modal particles and their influence on the narrative point of view. En *1st IATIS conference: Translation and the Construction of Identity*, Seul, South Korea, Agosto. International Association for Translation and Interecultural Studies (IATIS).

Zanettin, Federico, Silvia Bernardini, y Dominic Stewart, editores. 2003. *Corpora in translator education*. St. Jerome, Manchester, UK ; Northampton, MA.

Escopo *in situ**

Luiz Arthur Pagani
Universidade Federal do Paraná
arthur@ufpr.br

Resumo

No presente texto, apresentamos uma proposta alternativa para a análise da interação do escopo de expressões quantificadas, sem que as expressões precisem ser movidas. As interpretações são obtidas através de operadores de escopo, expressos por termos-λ puros; são precisos dois pares de operadores: um para a relação entre sujeito e predicado e outro para a ligação entre o verbo e seu objeto direto.

Palavras chave

Quantificadores, escopo, *in situ*

Abstract

In the present text, an alternative proposal is presented for the analysis of scope interaction among quantified expressions, without any movement. Both interpretations are obtained by scope operators, expressed by pure λ-terms — we need two pairs of operators: one for the relation between subject and predicate, and other for the link between the verb and its object.

Keywords

Quatifiers, scope, *in situ*

1 Introdução

A ambiguidade devido à interação entre quantificadores já é conhecida há muito tempo, na semântica, e pode ser encontrada em diversos manuais de introdução a esta disciplina, como Cann (1993, p. 180), de Swart (1998, p. 97), Chierchia e McConnell-Ginet (2000, p. 39), Oliveira (2001, p. 194), Chierchia (2003, p. 380) e Cançado (2005, p. 71). Um exemplo clássico desta ambiguidade é o da sentença *todo homem ama uma mulher*, que pode ser interpretada em duas situações:

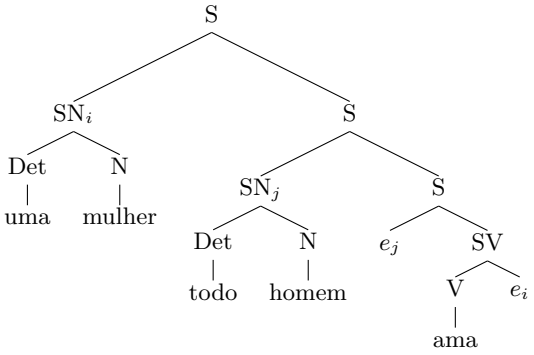


Figura 1: Árvore para escopo da quantificação existencial sobre a universal por deslocamento.

1. há uma única mulher que é amada por cada homem,
2. cada homem ama a sua respectiva mulher (no limite, uma mulher diferente para cada homem).

Na maioria destes manuais, esta ambiguidade é explicada através do alçamento dos quantificadores — com em Chierchia e McConnell-Ginet (2000, p. 157), Chierchia (2003, p. 373) e van Riemsdijk e Williams (1991, p. 194)).[1] Assim, cada uma destas interpretações seria representada pelas estruturas em árvore das Figuras 1 e 2.

No presente texto, apresento uma alternativa mais semântica (e até onde sei, inédita), pois os sintagmas quantificados não precisam ser

*Agradeço a Rui Marques, da Universidade de Lisboa, pela leitura atenta e pelos valiosos comentários, que me permitiram corrigir alguns erros e apontaram caminhos interessantes para continuar a investigação relatada aqui.

[1] Em Cann (1993, p. 186), fala-se de *quantifying in*; e em de Swart (1998, p. 97), fala-se ainda de "armazenamento de Cooper". Este primeiro e os outros mencionados no texto recorrem a uma solução essencialmente sintática, enquanto o segundo é uma solução mais computacional do que linguística. Outra alternativa para tratar o escopo sem movimentação de constituintes pode ser encontrada na DRT (Kamp e Reyle, 1993, ps. 279–304), onde se sugere basicamente uma solução através do relaxamento da ordem em que as regras de construção das representações são aplicadas; nesse sentido, esta também é uma solução computacional (foi o Rui Marques quem me lembrou desta solução na DRT, que eu havia esquecido de incluir numa versão anterior).

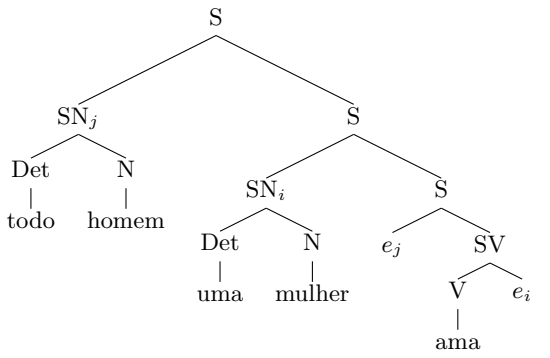


Figura 2: Árvore para escopo da quantificação universal sobre a existencial por deslocamento.

movidos (quantificação *in situ*) e o seu escopo é construído por operadores representados por termos-λ puros (nos quais não aparece nenhuma constante). Ainda que existam outras propostas para quantificação *in situ*, mesmo recorrendo a termos-λ puros, a minha proposta se distingue destas porque ela emprega apenas dois pares de operadores que atuam nas relações entre o sujeito e o predicado, e entre o verbo e o objeto direto; nas outras propostas, até onde pudemos avaliar, as soluções envolvem operações de ordem lexical, basicamente de promoção de tipo (*type-shifting*), sem nenhuma limitação de atuação (ou seja, correspondiam a esquemas infinitos de produção de operadores; minha proposta se restringe exclusivamente aos dois pares de operadores mencionados).

Nesse sentido, a alternativa proposta aqui deve ser entendida antes como um exercício formal, onde se apresenta um outro recurso para a representação da ambiguidade de escopo, e não como uma descrição da ambiguidade de escopo, nem como uma explicação para os fenômenos empíricos associados a ela. Assim, neste momento, não é relevante o fato de uma sentença com os quantificadores invertidos, como *uma mulher ama todos os homens*, apresentar ou não as mesmas duas leituras que *todo homem ama uma mulher*; no máximo, caso só exista uma única interpretação para *uma mulher ama todos os homens*, talvez se pudesse recorrer, por exemplo, a algum princípio que filtrasse a segunda leitura inadequada (o que ainda dependeria da determinação dos critérios empíricos para sua identificação); de qualquer maneira, esta é uma questão que exigiria uma pesquisa empírica, que ainda não foi executada, de teor distinto dos operadores sugeridos aqui apenas como ferramenta de formalização.[2]

[2]Devo novamente a Rui Marques a indicação desta questão, que não havia sido considerada incialmente. De qualquer maneira, gostaria ainda de chamar a atenção para o fato de que a simples inversão do sujeito e do objeto, na sentença (3), parece resultar numa sentença que causa algum estranhamento, ainda que não seja totalmente agramatical ou ininterpretável (pelo menos no português brasileiro): *uma mulher ama todo homem* (o mais natural seria dizer *uma mulher ama todos os homens*, como o próprio Rui preferiu nos seus comentários ao meu texto); todas estas observações parecem sugerir difuldades empíricas ainda maiores para esta generalização da inversão dos escopos.

S	$\rightarrow$	SN	SV
SN	$\rightarrow$	Det_{Gen}	N_{Gen}
SV	$\rightarrow$	V	SN

Tabela 1: Regras sintagmáticas.

Det_{masc}	$\rightarrow$	todo
Det_{fem}	$\rightarrow$	uma
N_{masc}	$\rightarrow$	homem
N_{fem}	$\rightarrow$	mulher
V	$\rightarrow$	ama

Tabela 2: Regras lexicais.

2 Recursos iniciais

Como vamos adotar uma semântica composicional, na qual o significado das expressões complexas é dado em função do significado das expressões mais simples que a compõem, levando ainda em consideração a maneira como essas expressões mais simples são combinadas, precisamos de um conjunto de regras sintáticas que nos diga como a sentença *todo homem ama uma mulher* está estruturada.

2.1 Sintaxe

Para lidar apenas com o exemplo apresentado, precisamos de uma sintaxe bastante simples, como a das Tabelas 1 e 2.

Com estas regras, podemos construir a árvore sintagmática de *todo homem ama uma mulher*, sem qualquer movimento, como se vê na Figura 3.

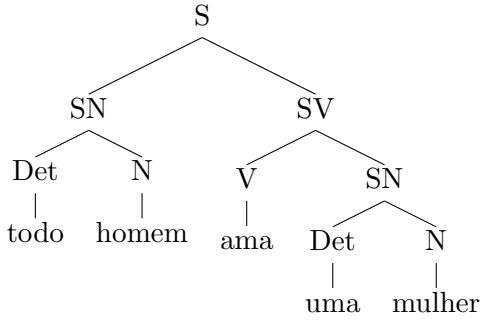


Figura 3: Árvore sem movimento para *todo homem ama uma mulher*.

Não há nada de controverso numa árvore como esta, além do fato evidente de que ela é apenas uma das muitas possibilidades estruturais para se construir sentenças em português, e portanto aquele conjunto de regras está muito longe de ser empiricamente exaustivo. No entanto, como nosso objetivo aqui não é empírico, mas sim o de apresentar uma ferramenta formal que pode vir a ser depois aplicada a outros fenômenos, mas que primeiro precisa ser integralmente compreendida, passaremos a apresentar as regras de interpretação semântica para esta pequena sintaxe.

2.2 Semântica

De acordo com a semântica formal, a interpretação das expressões de uma língua pode ser definida indutivamente, identificando a interpretação das unidades básicas (no caso das línguas naturais, estas unidades são os itens lexicais) e definindo, para cada regra sintagmática, o modo como os significados de suas partes interagem para resultar na interpretação do sintagma.

Para dar conta de *todo homem ama uma mulher*, vamos assumir os seguintes significados para cada um dos itens lexicais:

- $[\![\text{todo}]\!] = \lambda x_m.\lambda x_n.\forall x_o.((x_m\ x_o) \rightarrow (x_n\ x_o))$
- $[\![\text{uma}]\!] = \lambda x_m.\lambda x_n.\exists x_o.((x_m\ x_o) \wedge (x_n\ x_o))$
- $[\![\text{homem}]\!] = H$
- $[\![\text{mulher}]\!] = M$
- $[\![\text{ama}]\!] = A$

Para os determinantes *todo* e *uma*, os significados são aqueles adotados tradicionalmente e correspondem à noção de quantificador generalizado, em que a interpretação do determinante é uma função que toma dois predicados lógicos (primeiro a interpretação do nome comum ao qual ele se combina diretamente, e depois a interpretação de um predicado gramatical). Os nomes comuns *homem* e *mulher* denotam, como de costume, os conjuntos H, dos homens, e M, das mulheres (ou, mais precisamente, as funções características H e M — aplicadas ao universo do discurso, essas funções características separam respectivamente o conjunto dos homens e o conjunto das mulheres). Finalmente, o verbo *ama* denota a relação A, que se estabelece entre dois indivíduos de tal forma que um deles ama o outro (na notação que será empregada aqui, para dizermos que x ama y, escreveremos $((A\ y)\ x)$).[3]

[3]Há uma pequena variação em relação à notação mais comum $A(x, y)$; mas o formato escolhido se justifica pela comodidade para manipular as fórmulas que usaremos para a representação das interpretações.

Além dos significados dos itens lexicais, a interpretação de cada uma das regras sintagmáticas deve ser a seguinte:

- $[\![S]\!] = ([\![SN]\!]\ [\![SV]\!])$
- $[\![SN]\!] = ([\![Det]\!]\ [\![N]\!])$
- $[\![SV]\!] = \lambda x_m.([\![SN]\!]\ \lambda x_n.(([\![V]\!]\ x_n)\ x_m))$

A primeira regra nos informa que a interpretação do sintagma nominal sujeito é uma função que toma como argumento a interpretação do sintagma verbal e resulta na interpretação da sentença; ou seja, a interpretação da sentença resulta da aplicação funcional da interpretação do sintagma nominal sujeito à interpretação do sintagma verbal. A interpretação dos sintagmas nominais têm exatamente a mesma estrutura: o significado do determinante é uma função que toma como argumento o significado do nome comum e resulta na interpretação do sintagma nominal. A interpretação do sintagma verbal é um pouco mais complexa porque precisa lidar com as posições argumentais da relação denotada pelo verbo;[4] mas ela também é essencialmente a aplicação funcional da interpretação do sintagma nominal que é o objeto direto ao significado do verbo.[5]

Finalmente, vamos recorrer também a uma operação de redução. Ainda que não seja uma necessidade lógica, a redução-β permite a simplificação de certos termos-λ, de forma que a leitura das fórmulas fica bastante facilitada. A redução-β é parte de um dos três axiomas do cálculo-λ (Carpenter, 1997, p. 50) e pode ser definida da seguinte maneira:

$$(\lambda\alpha.\beta\ \gamma) = \beta^{\alpha \mapsto \gamma} \quad (1)$$

De acordo com esta definição, um termo $\lambda\alpha.\beta$ aplicado a um argumento γ é equivalente

[4]Uma alternativa que deixaria a interpretação do sintagma verbal mais simples envolveria uma modificação na atribuição lexical do verbo, que ficaria como $[\![\text{ama}]\!] = \lambda x_m.\lambda x_n.((A\ x_m)\ x_n)$; dessa maneira, a regra poderia ser reescrita mais simplesmente apenas como $[\![SV]\!] = ([\![SN]\!]\ [\![SV]\!])$. Como ambas as opções são completamente equivalentes, preferimos aqui manter os itens lexicais inalterados, já que o objetivo é manipular explicitamente as interpretaçoes das regras sintagmáticas.

[5]Ainda que esta regra de interpretação do sintagma verbal não seja comum nos manuais de semântica, uma forma parecida aparece nos manuais de linguística computacional em que se acrescenta capacidade interpretativa aos analisadores sintáticos, como Pereira e Shieber (1987, p. 91) e Covington (1994, p. 196). É neste último também que encontramos a inspiração para elaborar as árvores com nós anotados também com interpretações semânticas (Covington, 1994, p. 65); ainda que as apresentadas aqui sejam um pouco mais complexas.

ao termo β, mas substituindo nele todas as ocorrências livres da variável α pelo termo γ.[6] Assim, por exemplo, o termo

$$((\lambda x_1.\lambda x_2.((x_1\ x_2)\ x_2)\ R)\ a) \quad (2)$$

pode ser simplificado, em dois passos; primeiro para

$$(\lambda x_2.((R\ x_2)\ x_2)\ a) \quad (3)$$

e depois para

$$((R\ a)\ a) \quad (4)$$

Neste exemplo,

$$\lambda x_1.\lambda x_2.((x_1\ x_2)\ x_2) \quad (5)$$

é um operador que toma o predicado de dois argumentos R e o transforma num predicado reflexivo de um único argumento

$$\lambda x_2.((R\ x_2)\ x_2) \quad (6)$$

que, depois de aplicado ao argumento a, resulta em (4).

2.3 Construindo a interpretação

Para colocarmos em uso nossos itens lexicais e nossas regras de interpretação para as estruturas sintagmáticas, as suas variáveis precisam receber uma identificação para evitar casamentos indevidos. Por isso, os índices m, n e o devem ser substituídos por números inteiros que ainda não tenham sido usados na construção da interpretação.

Tomando esta precaução de não confundir a identidade das variáveis e aplicando as regras, podemos construir a árvore da Figura 4 para a interpretação de *todo homem ama uma mulher*, na qual o quantificador universal tem escopo mais amplo do que o existencial sem que haja qualquer deslocamento de constituintes.

Nela, como se vê depois da última redução-β, no nó relativo à sentença, obtemos a interpretação em (7).

$$\forall x_3.((H\ x_3) \rightarrow \exists x_6.(((A\ x_6)\ x_3) \wedge (M\ x_6))) \quad (7)$$

De acordo com esta fórmula, a sentença *todo homem ama uma mulher* recebe uma interpretação que poderia ser parafraseada como 'para todo indivíduo, se esse indivíduo é homem, então existe um indivíduo que é mulher e o primeiro indivíduo ama este segundo indivíduo'.

No entanto, com as regras propostas até aqui, esta é a única interpretação possível para *todo homem ama uma mulher*; a outra interpretação, na qual 'existe um indivíduo que é mulher e para todo indivíduo, se este indivíduo é homem, então o segundo indivíduo ama o primeiro indivíduo', não poderia ser construída.

3 Operadores de escopo

Para conseguirmos chegar também à segunda interpretação, a alternativa proposta aqui depende da postulação de quatro operadores de escopo (dois para a relação entre o verbo e o objeto direto, e dois para a relação entre o sujeito e o predicado), além da reformulação das regras de interpretação dos sintagmas.

Para manipular a inversão de escopo entre os quantificadores nas posições de sujeito e objeto, precisamos de um par de operadores para cada um dos escopos: um dos operadores do par cuida da relação entre o verbo e o objeto, enquanto o outro cuida da combinação do sujeito com o predicado. Os quatro operadores são apresentados na Tabela 3, de forma que nas colunas se diferenciem as funções sintáticas dos operadores (na segunda coluna da tabela, ficam os operadores de combinação da interpretação do verbo com a interpretação do objeto direto; e na terceira coluna ficam os operadores que combinam a interpretação do sujeito com a interpretação do predicado), e nas linhas se identifiquem as relações de escopo (na segunda linha, ficam os operadores que dão ao sujeito escopo mais amplo do que o do objeto direto; na terceira linha, temos os operadores que fazem com que o objeto direto tenha escopo maior do que o do sujeito).

Dessa maneira, os operadores na segunda linha da Tabela 3 serão responsáveis pela construção do escopo maior de *todo homem* em relação a *uma mulher*; os operadores da terceira linha, por sua vez, construirão o escopo amplo de *uma mulher* em relação a *todo homem*.

As novas regras para interpretação das estruturas sintagmáticas passam a ser as seguintes:

1. $[\![S]\!] = ((Escopo\ [\![SV]\!])\ [\![SN]\!])$
2. $[\![SN]\!] = ([\![Det]\!]\ [\![N]\!])$
3. $[\![SV]\!] = ((Escopo\ [\![V]\!])\ [\![SN]\!])$

Estas regras são mais uniformes do que as anteriores porque tanto na regra de interpretação de

[6] Existem ainda restrições para evitar o casamento indevido de variáveis, mas podemos evitá-las usando o recurso de exigir que todas as variáveis introduzidas sejam novas.

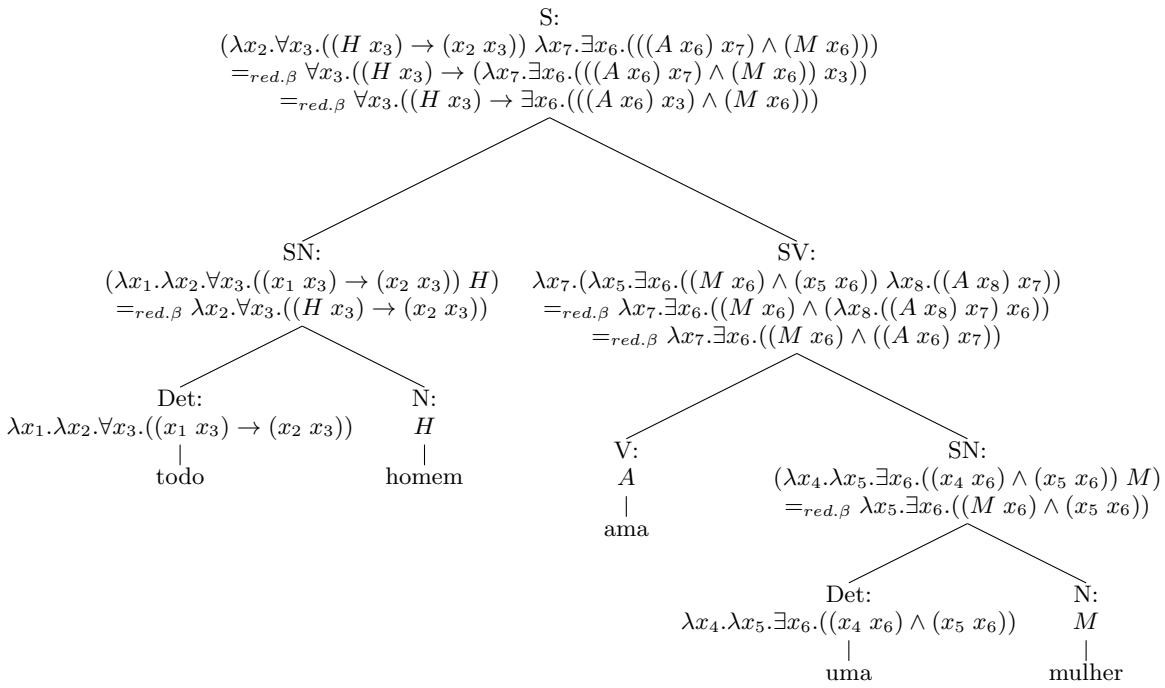


Figura 4: Única árvore para *todo homem ama uma mulher* sem os operadores de escopo.

Escopo	Entre Verbo e Objeto	Entre Sujeito e Predicado
Sujeito sobre Objeto	$\lambda x_m.\lambda x_n.\lambda x_o.(x_n\ \lambda x_p.((x_m\ x_p)\ x_o))$	$\lambda x_m.\lambda x_n.(x_n\ x_m)$
Objeto sobre Subjeito	$\lambda x_m.\lambda x_n.\lambda x_o.(x_n\ \lambda x_p.(x_o\ (x_m\ x_p)))$	$\lambda x_m.\lambda x_n.(x_m\ x_n)$

Tabela 3: Operadores de escopo

S quanto na de SV são os operadores que manipulam todos os posicionamentos, inclusive o dos argumentos do predicado de dois lugares do SV. Em ambos os casos, o operador de escopo corresponde semanticamente a uma função que toma a interpretação do V (no caso da interpretação do SV) ou do SV (no caso da interpretação de S) e resulta numa função que ainda vai tomar a interpretação de um SN para construir a interpretação do respectivo sintagma.

4 Derivações dos escopos

A partir das especificações acima, a interpretação em que o escopo do sujeito é maior do que o do objeto direto é construída de acordo com a árvore da Figura 5.

Nesta árvore, a construção da interpretação dos SNs é exatamente a mesma da árvore da seção 2.3; dessa maneira, a interpretação dos SNs não precisa ser comentada.

A diferença entre as árvores das Figuras 4 e 5 começa a ser percebida na construção da interpretação do SV, que envolve, além da interpretação do V e do SN-objeto (A e $\lambda x_5.\exists x_6.((M\ \ x_6) \wedge (x_5\ \ x_6))$, respectivamente), o operador de escopo $\lambda x_m.\lambda x_n.\lambda x_o.(x_n\ \ \lambda x_p.((x_m\ \ x_p)\ \ x_o))$. Como já havíamos empregado x_1, x_2 e x_3 para a interpretação de *todo*, e x_4, x_5 e x_6 para *uma*, e precisamos de variáveis novas para este operador de escopo, fazemos com que suas variáveis tenham a seguinte identidade: $\lambda x_7.\lambda x_8.\lambda x_9.(x_8\ \ \lambda x_{10}.((x_7\ \ x_{10})\ \ x_9))$.[7] Este operador toma como primeiro argumento a interpretação de *ama* — $(\lambda x_7.\lambda x_8.\lambda x_9.(x_8\ \ \lambda x_{10}.((x_7\ \ x_{10})\ \ x_9))\ \ A)$ — e, depois da redução-β, resulta em

[7] Como já dissemos, esse recurso serve para evitar que as variáveis acabem sendo indevidamente ligadas, como é exigido pelas teorias de demonstração de teoremas. Carpenter (1997, p. 156), por exemplo, falando das hipóteses, diz que "devemos garantir que as variáveis usadas nas hipóteses sejam novas, no sentido de que elas ainda não tenham sido empregadas em nenhuma outra hipótese usada anteriormente na derivação"; apesar de Carpenter mencionar apenas as hipóteses, como os itens lexicais introduzem variáveis na derivação, também era de se esperar que eles não fossem responsáveis por ligações indevidas.

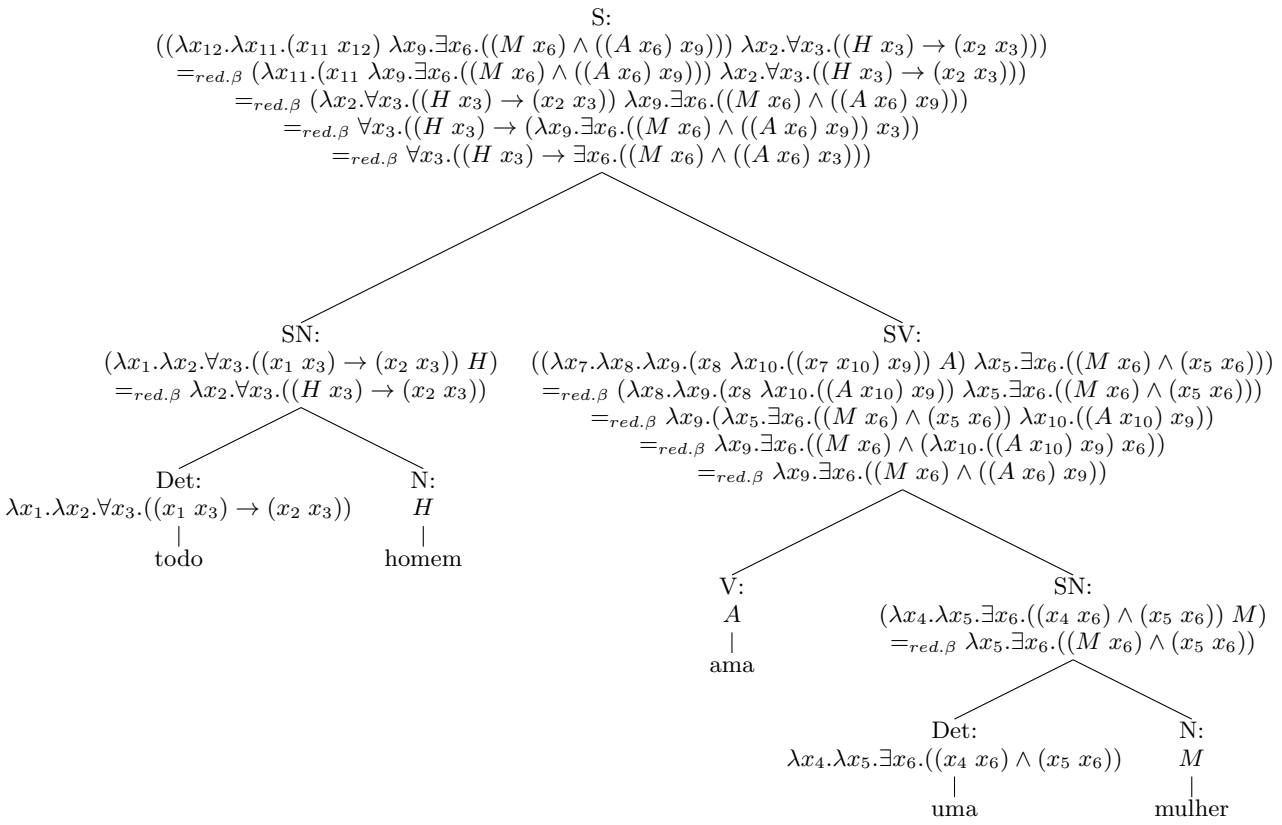


Figura 5: Árvore para escopo amplo do quantificador universal em *todo homem ama uma mulher* com operadores de escopo.

$\lambda x_8.\lambda x_9.(x_8\ \lambda x_{10}.((A\ x_{10})\ x_9))$. Na seqüência, o outro argumento tomado é a interpretação de *uma mulher* — $(\lambda x_8.\lambda x_9.(x_8\ \lambda x_{10}.((A\ x_{10})\ x_9))\ \lambda x_5.\exists x_6.((M\ x_6) \wedge (x_5\ x_6)))$ — que, depois de três reduções-β, resulta na interpretação do SV: $\lambda x_9.\exists x_6.((M\ x_6) \wedge ((A\ x_6)\ x_9))$. footnote-Tanto neste, quanto nos outros operadores de escopo, a ordem de combinação com os argumentos poderia ser outra (tomando primeiro a interpretação do SN e depois a do V ou a do SV), o que resultaria em termos-λ diferentes para representá-los; no entanto, se as relações adequadas forem mantidas, os resultados são exatamente os mesmos obtidos aqui. Portanto, como esta inversão, depois de compreendido o processo de abstração, é uma operação mecânica e trivial, não a discutiremos aqui.

Esta interpretação do SV é empregada depois, na construção da interpretação de S, porque o operador de escopo entre sujeito e predicado, depois de ter a identidade de suas variáveis estabelecida, a toma como primeiro argumento — $(\lambda x_{12}.\lambda x_{11}.(x_{11}\ x_{12})\ \lambda x_9.\exists x_6.((M\ x_6) \wedge ((A\ x_6)\ x_9)))$ — e, ao passar por uma redução-β, resulta em $\lambda x_{11}.(x_{11}\ \lambda x_9.\exists x_6.((M\ x_6) \wedge ((A\ x_6)\ x_9)))$. Em seguida, toma como segundo argumento a interpretação de *todo homem* — $(\lambda x_{11}.(x_{11}\ \lambda x_9.\exists x_6.((M\ x_6) \wedge ((A\ x_6)\ x_9)))\ \lambda x_2.\forall x_3.((H\ x_3) \rightarrow (x_2\ x_3)))$ — e, novamente depois de passar por três reduções-β, resulta na interpretação de S em que *todo homem* tem escopo maior do que *uma mulher*: $\forall x_3.((H\ x_3) \rightarrow \exists x_6.((M\ x_6) \wedge ((A\ x_6)\ x_3)))$.

Para a derivação do escopo do objeto sobre o sujeito, precisamos do outro par de operadores, e sua construção pode ser apresentada como na árvore da Figura 6.

Desta vez, o escopo do quantificador existencial do objeto direto sobre o quantificador universal do sujeito começa com a aplicação do operador $\lambda x_m.\lambda x_n.\lambda x_o.(x_n\ \lambda x_p.(x_o\ (x_m\ x_p)))$. Depois que suas variáveis são devidamente identificadas para evitar a ligação indevida, este operador, como o anterior, toma como argumento primeiro a interpretação do verbo — $(\lambda x_7.\lambda x_8.\lambda x_9.(x_8\ \lambda x_{10}.(x_9\ (x_7\ x_{10})))\ A)$ — e, após uma redução-β resulta em $\lambda x_8.\lambda x_9.(x_8\ \lambda x_{10}.(x_9\ (A\ x_{10})))$. A seguir, toma-se a interpretação do objeto direto como segundo argumento — $(\lambda x_8.\lambda x_9.(x_8\ \lambda x_{10}.(x_9\ (A\ x_{10})))\ \lambda x_5.\exists x_6.((M\ x_6) \wedge (x_5\ x_6)))$ — que, depois de passar pela mesma seqüência de três reduções-β, nos faz chegar à interpretação do SV: $\lambda x_9.\exists x_6.((M\ x_6) \wedge (x_9\ (A\ x_6)))$. (Neste ponto,

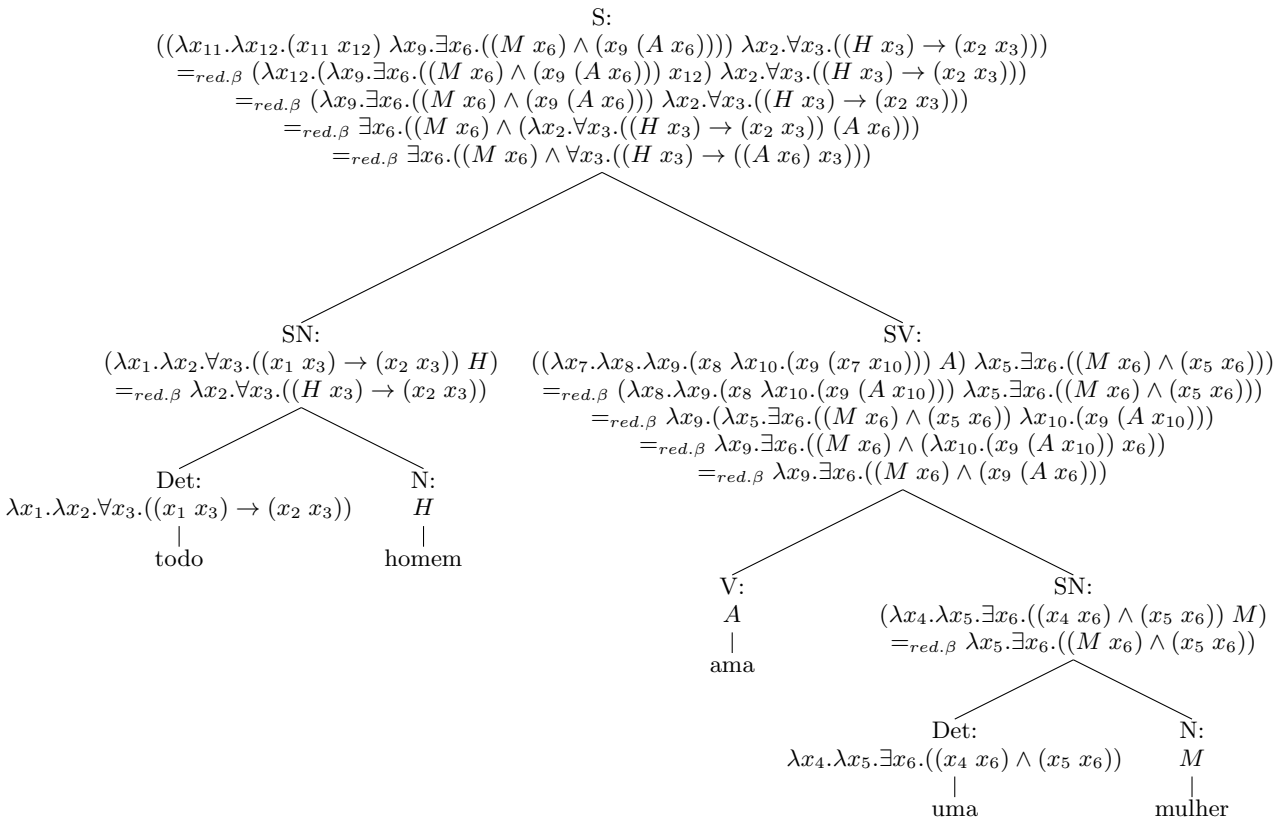


Figura 6: Árvore para escopo amplo do quantificador existencial em *todo homem ama uma mulher* com operadores de escopo.

convém lembrar a diferença entre a interpretação do SV que acabamos de "calcular" e a anterior — $\lambda x_9.\exists x_6.((M\ x_6) \wedge ((A\ x_6)\ x_9))$. Antes, a variável x_9 marcava o lugar de um argumento de tipo e (ou seja de um termo individual); agora, a mesma variável x_9 ocupa a posição de uma função (de tipo $\langle\langle e,t\rangle,t\rangle$) que tomará o predicado $(A\ x_6)$ como argumento.)

Com esta nova interpretação do SV como argumento do operador $\lambda x_m.\lambda x_n.(x_m\ x_n)$, com suas variáveis adequadamente identificadas, podemos começar a combinar o predicado com o sujeito — $(\lambda x_{11}.\lambda x_{12}.(x_{11}\ x_{12})\ \lambda x_9.\exists x_6.((M\ x_6) \wedge (x_9\ (A\ x_6))))$ — que se resolve como $\lambda x_{12}.(\lambda x_9.\exists x_6.((M\ x_6) \wedge (x_9\ (A\ x_6)))\ x_{12})$, com uma redução-$\beta$. A seguir, este termo toma como argumento a interpretação do sujeito — $(\lambda x_{12}.(\lambda x_9.\exists x_6.((M\ x_6) \wedge (x_9\ (A\ x_6)))\ x_{12})\ \lambda x_2.\forall x_3.((H\ x_3) \rightarrow (x_2\ x_3)))$ — e, mais uma vez com três reduções-β, resulta na interpretação da sentença com escopo do quantificador existencial sobre o universal: $\exists x_6.((M\ x_6) \wedge \forall x_3.((H\ x_3) \rightarrow ((A\ x_6)\ x_3)))$.

Dessa maneira, recorrendo aos quatro operadores de escopo, foi possível construir composicionalmente as duas fórmulas que representam a ambiguidade de escopo entre os quantificadores existencial e universal.

5 Interação dos operadores

Como os operadores de escopo são usados aos pares, um possível problema seria o controle da aplicação de um operador condicionado à aplicação do outro, para que eles não interagissem inadequadamente; no entanto, como os operadores tomam argumentos de tipos difetentes, e ainda geram expressões de tipos diferentes, esse controle decorre automaticamente da teoria de tipos, de forma que não precisa ser estipulado arbitrariamente.

Comecemos observando os contextos em que os operadores de escopo que combinam o verbo e o objeto direto atuam:

- $[\![SV]\!]^{\langle e,t\rangle} = ((Escopo\ [\![V]\!]^{\langle e,\langle e,t\rangle\rangle})\ [\![SN]\!]^{\langle\langle e,t\rangle,t\rangle})$
- $[\![SV]\!]^{\langle\langle\langle e,t\rangle,t\rangle,t\rangle} = ((Escopo\ [\![V]\!]^{\langle e,\langle e,t\rangle\rangle})\ [\![SN]\!]^{\langle\langle e,t\rangle,t\rangle})$

Ambos os operadores tomam como argumentos primeiro a interpretação do verbo e depois a do objeto direto; como o verbo transitivo denota uma relação de tipo $\langle e,\langle e,t\rangle\rangle$ e o quantificador generalizado sempre é do tipo $\langle\langle e,t\rangle,t\rangle$, o tipo do operador de escopo entre verbo e objeto direto será do tipo $\langle\langle e,\langle e,t\rangle\rangle,\langle\langle\langle e,t\rangle,t\rangle,?\rangle\rangle$, de forma que a identidade da incógnita ? depende do tipo que a interpretação do SV precisará ter.

Observando a árvore em que o escopo do universal é mais amplo do que o do existencial, constatamos que a interpretação do SV toma um indivíduo como argumento (e) para resultar numa proposição (t); portanto, seu tipo é $\langle e,t\rangle$ — o que faz com que o operador de escopo $\lambda x_m.\lambda x_n.\lambda x_o.(x_n\ \lambda x_p.((x_m\ x_p)\ x_o))$ tenha o tipo $\langle\langle e,\langle e,t\rangle\rangle,\langle\langle\langle e,t\rangle,t\rangle,\langle e,t\rangle\rangle\rangle$. Na árvore em que o existencial tem escopo sobre o universal, o tipo da interpretação do SV é $\langle\langle\langle e,t\rangle,t\rangle,t\rangle$, porque ela será uma função que tomará um quantificador generalizado (de tipo $\langle\langle e,t\rangle,t\rangle$) para resultar numa proposição (de tipo t); assim, o operador de escopo $\lambda x_m.\lambda x_n.\lambda x_o.(x_n\ \lambda x_p.(x_o\ (x_m\ x_p)))$ tem que ser do tipo $\langle\langle e,\langle e,t\rangle\rangle,\langle\langle\langle e,t\rangle,t\rangle,\ langle\langle\langle e,t\rangle,t\rangle,t\rangle\rangle\rangle$. Dessa maneira, os operadores de escopo que combinam o verbo e o objeto direto têm tipos distintos, basicamente em relação a seus resultados diferentes, por isso seus produtos não podem ser usados nos mesmos lugares um do outro.

Observemos agora os operadores de escopo que combinam sujeito e predicado:

- $[\![S]\!]^t = ((Escopo\ [\![SV]\!]^{\langle e,t\rangle})\ [\![SN]\!]^{\langle\langle e,t\rangle,t\rangle})$
- $[\![S]\!]^t = ((Escopo\ [\![SV]\!]^{\langle\langle\langle e,t\rangle,t\rangle,t\rangle})\ [\![SN]\!]^{\langle\langle e,t\rangle,t\rangle})$

Como se pode perceber, eles "herdam" tipos diferentes para a interpretação do SV, que será seu primeiro argumento, para depois tomar a interpretação do SN-sujeito, e finalmente resultarem numa proposição. Assim, o operador de escopo do universal sobre o existencial, $\lambda x_m.\lambda x_n.(x_n\ x_m)$, é do tipo $\langle\langle e,t\rangle,\langle\langle\langle e,t\rangle,t\rangle,t\rangle\rangle$. Já o operador para o existencial com escopo sobre o universal, $\lambda x_m.\lambda x_n.(x_m\ \ x_n)$, é do tipo $\langle\langle\langle\langle e,t\rangle,t\rangle,t\rangle,\langle\langle\langle e,t\rangle,t\rangle,t\rangle\rangle$.

Na Tabela 4, resumimos estas informações, colocando lado a lado todos os operadores e seus respectivos tipos, para facilitar a visualização de que seus tipos não permitiriam que eles fossem empregagos em outra ordem a não ser aquela utilizada nas duas árvores da seção anterior.

6 Conclusões

O objetivo deste texto era apenas o de apresentar os operadores de escopo que permitem as derivações das duas leituras em que os quantificadores da sentença *todo homem ama uma mulher* interagem, produzindo a clássica ambigüidade relatada em diversos manuais de semântica. Como o que se pretendia exclusivamente era apresentar uma ferramenta formal, a proposta não se constitui numa contestação das soluções transformacionalistas ou computacionais, nem tãopouco da DRT; da mesma maneira, o único exemplo apresentado apenas ilustra a questão, e portanto nem chega a tocar na questão empírica da abrangência desse tipo de ambiguidade.

Sobre isso, sabe-se que nem sempre dois quantificadores interagem para resultar numa ambiguidade de escopo (Morrill, 1994, p. 42); uma construção relativa, por exemplo, bloqueia a ambigüidade de escopo, pois o quantificador que aparecer dentro da relativa não pode ter escopo maior do que o quantificador fora dela. Assim, para o SN *todo homem que ama uma mulher*, não é possível a interpretação segundo a qual 'existe uma mulher tal que todo homem a ama'; sua única interpretação é a de 'para todo homem, existe uma mulher tal que ele a ama' . Como não tratamos da questão da subordinação, os operadores de escopo apresentados aqui não seriam capazes de gerar esse escopo amplo para o quantificador da relativa; mas, claro, tudo dependeria ainda da definição sobre como interpretar o pronome relativo.[8]

No entanto, seria impossível deixar de ressaltar um aspecto positivo dos operadores de escopo apresentados aqui, em relação às alterantivas de alçamento do quantificador (*quantifier raising*, (van Riemsdijk e Williams, 1991, p. 194)) e de introdução do quantificador (*quantifying-in*, (Morrill, 1994, p. 28)). Ainda que a introdução da quantificação seja pior do que o alçamento, já que a primeira permite a quantificação vazia, enquanto a segunda não,[9] ambas permitem igualmente uma proliferação de estruturas que não afetam a interpretação.

Além das árvores nas Figuras 2 e 4, podemos construir uma terceira árvore, como na Figura 7, também através do alçamento do quantificador. No entanto, nesta terceira possibilidade, a interpretação seria exatamente a mesma das duas anteriores, com o quantificador universal com escopo sobre o existencial.

Com os operadores de escopo apresentados aqui, essa proliferação não acontece, porque

[8] Mas as chamadas ilhas podem não ser os únicos fatores que restringem a ambiguidade de escopo. Características como o tamanho do SN (os famosos SNs pesados que, segundo uma tradição gerativista mais antiga, eram responsáveis pela posposição do SN) e a especificidade (SN indefindos aos quais se acrescentam modificadores que tornam sua interpretação mais específica tendem a ter uma leitura referencial) também podem estar envolvidos na restrição das interações dos escopos.

[9] Na quantificação vazia, segundo o exemplo de Morrill (1994, p. 37), a sentença *Pedro caminha* pode ser resultado da combinação dela própria com o quantificador *toda mulher*, de forma que ela significaria algo como 'para todo indivíduo, se esse indivíduo é mulher, então Pedro caminha', o que é claramente uma falha.

Operador	Tipo 1o. arg.	2o. arg.	res.
$\lambda x_m.\lambda x_n.(x_n\ x_m)$	$\langle\ \langle e,t\rangle$	$,\langle\ \langle\langle e,t\rangle,t\rangle$	$,\ t\ \rangle\rangle$
$\lambda x_m.\lambda x_n.(x_m\ x_n)$	$\langle\ \langle\langle\langle e,t\rangle,t\rangle,t\rangle$	$,\langle\ \langle\langle e,t\rangle,t\rangle$	$,\ t\ \rangle\rangle$
$\lambda x_m.\lambda x_n.\lambda x_o.(x_n\ \lambda x_p.((x_m\ x_p)\ x_o))$	$\langle\ \langle e,\langle e,t\rangle\rangle$	$,\langle\ \langle\langle e,t\rangle,t\rangle$	$,\ \langle e,t\rangle\ \rangle\rangle$
$\lambda x_m.\lambda x_n.\lambda x_o.(x_n\ \lambda x_p.(x_o\ (x_m\ x_p)))$	$\langle\ \langle e,\langle e,t\rangle\rangle$	$,\langle\ \langle\langle e,t\rangle,t\rangle$	$,\ \langle\langle\langle e,t\rangle,t\rangle,t\rangle\ \rangle\rangle$

Tabela 4: Identificação dos tipos dos operadores de escopo

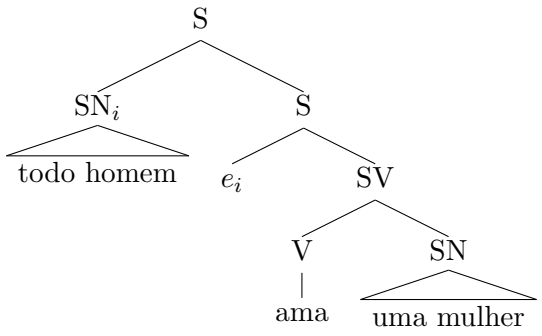


Figura 7: Mais uma árvore de escopo amplo do quantificador universal com deslocamento.

as duas interpretações são obtidas a partir da mesma estrutura sintática.

A presente proposta apresenta algumas semelhanças com soluções anteriores, como a de Hendriks (1993), a de Jacobson (1996) e a de Heim e Kratzer (1998). Nestas, contudo, a interação entre os escopos dos quantificadores era obtida por uma operação lexical de promoção de tipo, que pode ser aplicada arbitrariamente aos seus próprios resultados, o que gera uma proliferação infinita e desnecessária de escopagem. Aqui os dois pares de operadores de escopo geram apenas as duas interpretações; nesta concepção, a interação de escopo não é apresentada como um fenômeno que afeta indiscriminadamente qualquer posição gramatical: ela atinge apenas as posições designadas pelos operadores. Se, além da subordinada, os sintagmas preposicionados também forem "ilhas" para a interação de escopo, apenas os núcleos dos sujeitos e dos objetos diretos devem oferecer possibilidade de interação de escopo; os objetos indiretos, os complementos e os adjuntos adnominais e também os adjuntos adverbiais, preposicionados, não devem permitir que os escopos de seus possíveis quantificadores interajam com outros quantificadores, tornando desnecessária a postulação de outros operadores. Porém, evidentemente, esta é uma previsão empírica que não foi efetivamente testada aqui, e ainda aguarda uma investigação mais detalhada.

Finalmente, gostaria de ressaltar que, nesse sentido, a presente proposta parece ser inédita. Os termos-λ puros apresentados aqui como operadores de escopo não foram retirados de nenhuma proposta conhecida (nem mesmo inspirados em qualquer uma delas, apesar de suas semelhanças superficiais). Ainda que, em essência, este tenha sido apenas um exercício de formalização e de dedução dos operadores de escopo, suas consequências para a análise de fenômenos linguísticos (que, do ponto de vista empírico, seria o mais interessante para um linguista) parecem ser promissoras à medida que oferecem uma alternativa com menos "efeitos colaterais" (como a multiplicação de estruturas com mesma interpretação), apesar dos custos formais; estes, no entanto, poderiam ser facilmente implementados computacionalmente.

Referências

Cann, Ronnie. 1993. *Formal Semantics – An Introduction*. Cambridge University Press, Cambridge.

Cançado, Márcia. 2005. *Manual de Semântica – Noções Básicas e Exercícios*. Editora UFMG, Belo Horizonte.

Carpenter, Bob. 1997. *Type-Logical Semantics*. The MIT Press, Cambridge, MA.

Chierchia, Gennaro. 2003. *Semântica*. Editora da Unicamp & Editora da UEL, Campinas & Londrina. Traduzido por Luiz Arthur Pagani, Lígia Negri & Rodolfo Ilari.

Chierchia, Gennaro e Sally McConnell-Ginet. 2000. *Meaning and Grammar – An Introduction to Semantics*. The MIT Press, Cambridge, MA, second edition.

Covington, Michael A. 1994. *Natural Language Processing for Prolog Programmers*. Prentice Hall, Englewood Cliffs.

de Swart, Henriëtte. 1998. *Introduction to Natural Language Semantics*. CSLI, Stanford.

Heim, Irene e Angelika Kratzer. 1998. *Semantics in Generative Grammar*. Wiley-Blackwell, Oxford.

Hendriks, Herman. 1993. *Sudied Flexibility – Categories and Types in Syntax and Semantics*. Tese de doutoramento, Institute for Logic, Language and Computation, Amsterdam.

Jacobson, Pauline. 1996. The syntax/semantics interface in categorial grammar. Em Shalom Lappin, editor, *The Handbook of Contemporary Semantic Theory*. Blackwell, Oxford, pp. 89–116.

Kamp, Hans e Uwe Reyle. 1993. *From Discourse to Logic – Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Kluwer, Dordrecht.

Morrill, Glyn V. 1994. *Type Logical Grammar*. Kluwer, Dordrecht.

Oliveira, Roberta Pires de. 2001. *Semântica Formal*. Mercado das Letras, Campinas.

Pereira, Fernando C. N. e Stuart M. Shieber. 1987. *Prolog and Natural Language Analysis*. CSLI, Stanford.

van Riemsdijk, Henk e Edwin Williams. 1991. *Introdução à Teoria da Gramática*. Martins Fontes, São Paulo. Traduzido por Miriam Lemle, Maria Angela Botelho Pereira & Marta Coelho.

Tecnologias

Desenvolvimento de Aplicações em Perl com FreeLing 3

Alberto Simões
Centro de Estudos Humanísticos
Universidade do Minho
ambs@ilch.uminho.pt

Nuno Carvalho
Departamento de Informática
Universidade do Minho
narcarvalho@di.uminho.pt

Resumo

O FreeLing é uma ferramenta para processamento de linguagem natural, em especial para análise morfossintáctica e cálculo de árvores de dependências. Embora a escolha de implementação em C++ seja relevante pela eficiência, torna complicado o desenvolvimento de pequenas ferramentas. Além disso, a interface Perl disponibilizada com o próprio FreeLing não é mais que um mapeamento directo da API C++ para Perl, o que não é o mais adequado.

Este artigo apresenta as decisões de implementação do módulo Perl `FL3`, e discute como esta interface torna simples a escrita de pequenos processadores de linguagem natural em Perl.

Palavras chave

FreeLing3, Perl

Abstract

FreeLing is a tool for processing natural languages, especially for morphological analysis and computation of dependency trees. Although C++ is a suitable language to implement this kind of tool given its efficiency, it makes it difficult to develop small tools. Also, the Perl interface available with the FreeLing package is not much more than a simple map from the C++ API to Perl, which isn't the most appropriate.

This article we presents some decisions made to implement the Perl module `FL3`, and discusses how this interfaces makes it easy to write small natural language processors in Perl.

Keywords

FreeLing3, Perl, Processamento de Linguagem Natural

1 Introdução

O FreeLing[1] (Padró, 2011; Padró et al., 2010) é uma biblioteca para a construção de analisadores (morfológicos, sintácticos, e outros) multilingues. A versão 3 inclui suporte para UTF-8 e um leque interessante de línguas, das quais realçamos as principais línguas ibéricas, o inglês e o russo.

A biblioteca é composta por classes que abstraem componentes linguísticos, como sejam parágrafos, frases, palavras ou mesmo análises morfológicas, que são usadas por outras classes que implementam algoritmos de análise morfológica com suporte para detecção de termos multipalavra (datas, números, locuções e entidades mencionadas), algoritmos de *tagging*, um baseado em *Relax* e um outro baseado em cadeias de Markov (HMM), um algoritmo de *parsing*, baseado em Charts; um algoritmo de cálculo de dependências (Atserias, Comelles e Mayor, 2005), entre outros.

O FreeLing está escrito em C++, o que lhe confere eficiência, mas que dificulta o desenvolvimento rápido de pequenas ferramentas, que se implementam mais rapidamente utilizando linguagens de programação ditas de *scripting*. A integração desta biblioteca em ferramentas já desenvolvidas noutras linguagens também se pode tornar complicada. Por estas razões, o FreeLing disponibiliza um conjunto de modelos para a geração de interfaces noutras linguagens, como Perl, Python ou Java. Estes modelos são usados pela ferramenta SWIG[2] (Simplified Wrapper and Interface Generator) para a geração de uma API (Application Programmer Interface) para cada uma dessas linguagens.

Esta abordagem, baseada em SWIG, permite ao programador da biblioteca a definição de interfaces para diferentes linguagens de uma forma rápida e uniforme, mas a API obtida é apenas um mapeamento directo entre as classes e métodos do FreeLing para a linguagem de destino. Esta interface nem sempre é a mais versátil, já que habitualmente não tira partido das funcionalidades nem filosofia da linguagem de destino.

Estas razões levaram à implementação de um módulo Perl (`Lingua::FreeLing3`) que abstrai a

[1] O FreeLing está disponível gratuitamente a partir de `http://nlp.lsi.upc.edu/freeling/`

[2] http://www.swig.org/

interface gerada pelo SWIG. Este módulo torna o desenvolvimento de ferramentas em Perl usando o FreeLing mais simples e rápido.

Este artigo está organizado como se segue: em primeiro lugar é explicada a arquitectura do módulo, e como este interage com a biblioteca FreeLing. Posteriormente apresentam-se algumas ferramentas desenvolvidas utilizando este módulo, juntamente com versões simplificadas do código Perl que as implementa[3]. É importante a apresentação deste código, já que só deste modo se poderá perceber como foi feita a abstracção da biblioteca FreeLing. Finalmente, apresentam-se algumas conclusões e planos de trabalho futuro.

2 Arquitectura do `Lingua::FreeLing3`

O desenvolvimento do `Lingua::FreeLing3` foi feito com base na interface obtida aplicando o modelo SWIG disponibilizado, que gera um módulo Perl que invoca funções de cola em C (Jenness e Cozens, 2002) para a ligação com a biblioteca. O `Lingua::FreeLing3` inclui um conjunto de módulos (ou classes) um para cada tipo de objecto ou ferramenta disponibilizada pelo FreeLing. Cada uma destas classes independentes em Perl define uma API mais sucinta e de alto nível (Dominus, 2005).

A figura 1 apresenta as várias camadas e de que forma elas interagem. As duas primeiras camadas correspondem à biblioteca FreeLing, e à API gerada pelo SWIG. Na camada seguinte estão os vários módulos de *objectificação* da API gerada pelo SWIG, bem como alguma gestão de memória que não é feita convenientemente pelo código gerado. Finalmente,uma classe de alto nível (`FL3`) para permitir o uso elegante dos vários algoritmos disponíveis.

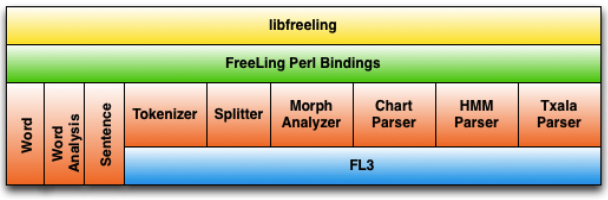


Figura 1: Níveis de Indirecção do FL3.

O módulo `FL3` simplifica a criação de processadores que tiram partido dos vários algoritmos de análise, usando um conjunto de opções por omissão.

[3]É importante salientar que se espera que o leitor tenho algum conhecimento do uso da linguagem de programação Perl, já que de outro modo se tornará difícil compreender completamente os exemplos apresentados. Além disso, os exemplos foram propositadamente simplificados em alguns aspectos, que não têm que ver com a interacção com a biblioteca FreeLing3.

3 Desenvolvimento com `Lingua::FreeLing3`

De modo a ilustrar o desenvolvimento de aplicações de processamento de linguagem natural usando o `Lingua::FreeLing3`, e o seu módulo de alto nível `FL3`, apresentam-se um conjunto de pequenas ferramentas úteis, e a sua implementação em Perl. Estas ferramentas não fazem parte do módulo `Lingua::FreeLing3`, mas estão disponíveis num módulo de ferramentas e utilidades: `Lingua::FreeLing3::Utils`.

3.1 N-Gramas

Em PLN um n-grama consiste num conjunto de sequências de n símbolos (tipicamente designados por *tokens*) consecutivos, calculados a partir de um texto. Apesar de estes *tokens* poderem ser de vários tipos, o mais comum é usarem-se n-gramas de caracteres ou de palavras, sendo estes últimos os mais usados para a criação de modelos de língua, associando o cálculo da probabilidade de ocorrência do respectivo n-grama no texto. A título de exemplo considere-se a seguinte frase: *"E o tempo responde ao tempo que o tempo tem tanto tempo quanto tempo o tempo tem."*. Podemos dividir a frase em 18 palavras (*tokens*), habitualmente designadas por formas. O cálculo de unigramas para este exemplo, bem como as respectivas probabilidades, são apresentados na tabela 1.

Unigramas	Ocor.	Probabilidade	
tempo	6	$P(tempo) = 6/18$	33%
o	3	$P(o) = 3/18$	17%
tem	2	$P(tem) = 2/18$	11%
E	1	$P(E) = 1/18$	6%
responde	1	$P(responde) = 1/18$	6%

Tabela 1: Unigramas e probabilidades associadas.

Para o cálculo das unigramas, o primeiro passo é dividir a frase em palavras (atomização ou *tokenization*), e o segundo, contar a ocorrência de cada uma das palavras. As respectivas probabilidades são calculadas dividindo o número de ocorrências pelo número total de palavras.

Efectuando o mesmo exercício mas para calcular bigramas obtemos alguns exemplos da tabela 2. O processo é similar, mas os *tokens* são agrupados dois a dois, e as respectivas probabilidades são calculados usando o teorema de Bayes.

Este processo é aplicado de forma semelhante para outros valores de n (Jurafsky et al., 2000).

Bigramas	Ocor.	Probabilidade	
o tempo	3	$P(tempo\|o) = 3/3$	100%
tempo tem	2	$P(tem\|tempo) = 1/3$	33%
tem tanto	1	$P(tanto\|tem) = 1/2$	50%

Tabela 2: Bigramas e probabilidades condicionadas associadas.

O cálculo de n-gramas é simples desde que seja possível calcular a sequência de palavras que compõem o texto. A tarefa complexa passa a ser calcular esta sequência de palavras, dado ser necessário lidar com uma série de problemas de solução não trivial: hifenização, sinais de pontuação, abreviaturas, etc.

O `FreeLing3` permite obter de forma relativamente fiável um conjunto de palavras a partir de um texto. Por exemplo, para obter a lista de *tokens* que formam o texto podemos simplesmente executar:[4]

```
my $tokens = tokenizer->tokenize($text);
```

A lista de símbolos torna-se disponível, e basta então percorrer essa sequência com uma janela deslizante de tamanho n para calcular uma lista de n-gramas.

Listagem 1: Cálculo de n-gramas.

```
# para todos os tokens da sequencia
for $c (0 .. @tokens - $n + 1)
{
  # calcular um n-gram com n elementos
  my @ngram = @$tokens[$c .. $c+$n-1];

  # calcular uma string...
  my $ngram = join(" ", @ngram);

  # incrementar o número de ocorrências
  $ngrams->{$ngram}{count}++;
}
```

Uma vez a lista de n-gramas e as respectivas ocorrências calculadas, o cálculo das probabilidades de cada um é apenas uma questão aritmética.

É bastante comum utilizar no cálculo de n-gramas os símbolos especiais `<s>` e `</s>` para marcar o início e fim de frase respectivamente, bastante úteis para rever quais as palavras com maior probabilidade nessas circunstâncias. Utilizando o `FreeLing3` é bastante fácil de produzir este efeito, uma vez que é possível a partir de uma lista de *tokens* obter uma lista de frases. A este processo chama-se normalmente segmentação (*segmentation*).

[4] É possível especificar a língua a usar passando-o como parâmetro: `tokenizer('en')->tokenize($text)`. Esta interface é semelhante para todas as outras funcionalidades do módulo.

Esta tarefa pode mais uma vez ser delegada para o `FreeLing3`. A partir de uma lista de *tokens* o FreeLing é capaz de os agrupar em frases:

```
my $sentences = splitter->split($tokens)
```

A listatem 2 ilustra o código necessário para implementar este processo.

Listagem 2: Adicionar símbolos de início e fim de frase.

```
# calcular tokens
my $tokens = tokenizer->tokenize($txt);

# calcular frases
my $sentences = splitter->split($tokens);

# para cada frase
foreach my $s (@$sentences) {
  # construir nova frase
  $s = sentence(word('<s>'),
                @$s,
                word('</s>'));
}
```

O resto do cálculo dos n-gramas é feito de modo análogo ao descrito anteriormente. A maior diferença nos resultados é que se passa a obter os novos *tokens*. Para o exemplo anterior passa a existir o bigrama "`<s>` O," que representa a palavra "O" no início da frase.

As operações descritas nesta seção foram implementadas numa ferramenta chamada `fl3-ngrams`. Segue-se um exemplo de execução para cálculo de bigramas:

```
$ fl3-ngrams -n 2 input.txt
# n-gram    count   prob
tempo o     1       0.16666667
tempo tem   2       0.33333333
E o         1       1.00000000
(...)
```

3.2 Análises Morfológicas

Não é uma aplicação muito habitual e, possivelmente, não muito útil por si só. Serve, no pior do casos, para consultar o dicionário de análise morfológica. Pretende-se uma aplicação que, dada uma palavra (ou uma sequência delas), nos permita obter todas as análises morfológicas possíveis para cada palavra. Ou seja, esta análise será não desambiguada.

No caso concreto desta aplicação não são precisas as fronteiras das frases, apenas de converter o texto num conjunto de palavras que possam ser analisadas. O código que se segue faz o processamento linha a linha, invocando a função `word_analysis` para obter as análises de cada pa-

lavra[5].

Listagem 3: Processamento de ficheiro para Análise Morfológica de Palavras

```
# processar cada linha
while (my $l = <>) {
  # atomizar a linha
  my $wrds = tokenizer->tokenize($l);

  # analisar cada palavra
  my @ws = word_analysis(@$wrds);

  # apresentar resultados
  while (@ws) {
    my $w = shift @$wrds;
    my $a = shift @ws;

    # apresentar forma da palavra
    print $w->form;

    # apresentar lema e POS possíveis
    for my $x (@$a) {
      print " [$x->{lemma}, $x->{tag}]"
} } }
```

Além disso, não se pretende que o Analisador Morfológico use informação de contexto e tente, por exemplo, fazer reconhecimento de entidades mencionadas ou locuções. Logo, é necessário desligar todas essas opções na função de cálculo de análises.

Listagem 4: Cálculo das Análise Morfológica

```
sub word_analysis {
  my @words = @_;

  # desactivar detecção de multipalavras
  my %conf = (
    ProbabilityAssignment  => 'no',
    QuantitiesDetection    => 'no',
    MultiwordsDetection    => 'no',
    NumbersDetection       => 'no',
    DatesDetection         => 'no',
    OrthographicCorrection => 'no',
    NERecognition          => 'no'
  );

  # criar frase artificial para analisar
  my $a = morph(%conf)->analyze(
     [ sentence(@words) ]
  );

  # converter cada conjunto de análises
  # numa lista de facetas
  return map {
    $_->analysis(FeatureStructure => 1)
  } $a->[0]->words;
}
```

3.3 NLGrep

Uma das aplicações típicas de corpos anotados é a pesquisa de concordância, procurando por determinado fenómeno linguístico, ou apenas para analisar o contexto de determinada palavra ou forma morfológica. Este processo é simples de se realizar depois de o corpo estar devidamente processado e disponível num motor adequado, como por exemplo o IMS Open Corpus Workbench (Christ, 1994).

Mas pode ser útil a pesquisa rápida sobre um texto não anotado, realizando anotação dinamicamente. Cada frase é anotada e o padrão indicado é procurado. Se for satisfeito, é apresentado o fragmento da frase que está de acordo com o padrão.

É certo que esta abordagem não é a melhor se o objectivo for realizar várias pesquisas sobre o mesmo texto, já que o texto será anotado para cada procura. Para resolver este problema é possível a criação de uma *cache* com o documento anotado, para ser usada em futuras execuções de modo a poupar tempo.

A linguagem de padrões implementada é relativamente simples, suportando apenas três tipos de condição por palavra: pesquisa de palavra exacta (`=palavra`); pesquisa por lema (`~lema`); ou pesquisa por (prefixo de) faceta morfológica (usando a etiqueta EAGLES `NCMS`, nome comum, masculino singular). Além disso, é possível usar o carácter sublinhado (`_`) como posição a associar a qualquer palavra (*wildcard*).

De seguida apresenta-se um exemplo do uso desta ferramenta, para a língua portuguesa, procurando num livro do Projecto Gutenberg. A expressão de pesquisa usada corresponde a uma forma do verbo "*ser*", seguida de um qualquer verbo, seguido da palavra "*o*", seguida de qualquer outra palavra:[6]

```
$ fl3-nlgrep pg33056.txt ~ser V =o _
era levantar o edificio
```

Embora extremamente simples, esta aplicação permite obter alguns resultados interessantes, como por exemplo, verificar que adjectivos são habitualmente usados com conjunções:

```
$ fl3-nlgrep -l pt pg33056.txt ~ser A C A
era grosso e baixo
era excellente e detestavel
é pura e severa
Sou exclusivo e pessoal
era orgulhoso e fraco
é independente e superior
era grande e vistosa
era justo nem bonito
é trivial e chocho
era restricta e mansa
```

[5]Neste momento a análise morfológica obtida é a etiqueta EAGLES usada no dicionário.

[6]Esta pesquisa concreta para o texto relativamente pequeno teve apenas um acerto.

A implementação desta ferramenta é igualmente simples. Para além do processamento típico já usado nas ferramentas anteriores (atomização, segmentação e anotação morfológica), o `fl3-nlgrep` adiciona uma nova camada de processamento usando um etiquetador (*tagger*). O FreeLing tem suporte para dois algoritmos de etiquetação, um baseado em cadeias de Markov (HMMTagger) e um outro baseado em Relax (RelaxTagger).

Listagem 5: Implementação do nlGrep

```
# inicializar analisador morfológico
morph(ProbabilityAssignment  => 'yes',
      QuantitiesDetection    => 'no',
      MultiwordsDetection    => 'no',
      NumbersDetection       => 'no',
      DatesDetection         => 'no',
      OrthographicCorrection => 'no',
      NERecognition          => 'no');

# abrir ficheiro a processar
open my $fh, "<:utf8", $filename;

# processar cada linha/frase
while (my $l = <$fh>) {
  my ($tokens, $frases);

  # de texto obter palavras
  $tokens=tokenizer->tokenize($l);
  # de palavras agrupar em frases
  $frases=splitter->split($tokens);
  # anotar morfologicamente
  $frases=morph->analyze($frases);

  # etiquetar usando cadeiras Markov
  $frases = hmm->tag($frases);

  # para cada frase
  for my $frase (@$frases) {
    my @words = $frase->words;

    # janela deslizante
    while (@words > @query) {
      if (match(\@words, \@query)) {
        show_match(@words[0..$#query])
      }
      shift @words;
} } }

# imprime as palavras.
sub show_match {
  print join(" ",map{$_->form} @_),"\n"
}

# verifica palavras contra expressão
# de pesquisa
sub match {
  for my $i (0 .. $#query) {
    # ignorar palavra se wildcard
    next if $query->[$i] eq "_";

    # se procuramos palavra exacta
    if ($query->[$i] =~ /^=(.*)$/) {
      if ($1 ne $words->[$i]->lc_form) {
        return 0
    } }
    # se procuramos por lema
    elsif ($query->[$i] =~ /^~(.*)$/) {
      if ($1 ne $words->[$i]->lemma) {
        return 0
    } }
    # caso contrário, etiqueta POS
    else {
      my $tag = $words->[$i]->tag;
      if ($tag !~ /^$query->[$i]/i) {
        return 0
  } } }
  return 1;
}
```

3.4 Extractor de EM

O último exemplo que aqui se apresenta tira partido de uma das funcionalidades do analisador morfológico do FreeLing, que é a detecção de multi-palavras, sejam quantidades, números, datas, locuções, ou genericamente, nomes próprios. Para além disso, o FreeLing incorpora um classificador de entidades mencionadas que é capaz de distinguir entre nomes próprios de pessoas, de organizações e de locais geográficos (bem como uma classe genérica, para todos outros tipos de classificação).

É importante realçar que nos exemplos apresentados não estamos a tentar demonstrar a qualidade, ou falta dela, do FreeLing. Note-se que ao aplicar várias ferramentas em cascata a percentagem de erro aumenta, como bola de neve. Além disso, os textos exemplos são de português antigo, e os ficheiros para detecção de entidades para a língua Portuguesa podem ainda levar bastantes melhorias.

As camadas de processamento utilizadas nesta ferramenta incluem, tal como nas anteriores, o atomizador e o segmentador. O analisador morfológico, com o módulo de detecção de nomes ligado e, finalmente, o módulo de classificação de entidades.

Listagem 6: Extracção de EM

```
# inicializar analisador morfológico
morph(ProbabilityAssignment  => 'yes',
      QuantitiesDetection    => 'no',
      MultiwordsDetection    => 'yes',
      NumbersDetection       => 'no',
      DatesDetection         => 'no',
      OrthographicCorrection => 'no',
      NERecognition          => 'yes');

# nomes das classes
my %classes = (NP00SP0 => 'Person',
               NP00G00 => 'Geographical',
               NP00O00 => 'Organization',
               NP00V00 => 'Others');

open my $fh, "< :utf8", $filename;
```

```
# processar cada linha...
while (my $l = <$fh>) {
  $tokens = tokenizer->tokenize($l);
  $frases = splitter->split($tokens);
  $frases = morph->analyze($frases);

  # classificar as entidades
  $frases = nec->analyze($frases);

  # fazer estatísticas das classificações
  for my $frase (@$frases) {
    for my $word ($frase->words) {

      # processar multi-palavras das
      # classes relevantes
      if ($word->is_multiword &&
          exists($classes{$word->tag})) {

        my $class = $classes{$word->tag};
        my $fw = $word->get_mw_words();
        $counts{$fw}{_}++;
        $counts{$fw}{$class}++;
      }
    }
  }
}

# imprimir resultados...
for my $mw (keys %counts) {
  print $mw;
  for my $clss (keys %{$counts{$mw}}) {
    next if $clss eq "_";
    printf "\t$clss (%.4f)",
      $counts{$mw}{$clss}/$counts{$mw}{_}
  }
  print "\n";
}
```

De seguida apresentam-se alguns exemplos. Note-se que por falta de espaço removeram-se alguns falsos positivos[7].

```
$ fl3-ner -l pt pg33056-pt.txt
João Braz         Per (0.80) Org (0.20)
Sacco do Alferes  Geo (1.00)
Sophia            Per (0.91) Geo (0.09)
Santa Thereza     Per (1.00)
Petropolis        Org (0.50) Geo (0.50)
Oriente           Geo (1.00)
Julia Costinha    Per (1.00)
Macbeth           Oth (1.00)
Joaquim           Per (1.00)
Luthero           Per (1.00)
Israel            Geo (1.00)
```

4 Conclusões e trabalho futuro

Este artigo apresenta o módulo Perl `Lingua::FreeLing3`, como método de interagir com a biblioteca FreeLing, e como, de forma simples, é possível implementar ferramentas úteis e relevantes.

Neste momento, estas quatro ferramentas estão a ser melhoradas e agrupadas num outro módulo Perl, de nome `Lingua::FreeLing3::Utils`, com o objectivo de disponibilizar algumas ferramentas básicas de PLN que actualmente não estão disponíveis para a linguagem Perl.

Agradecimentos

O trabalho aqui apresentado foi parcialmente financiado pelo projecto da Fundação para a Ciência e Tecnologia PTDC/CLE-LLI/108948/2008.

Os autores também agradecem a paciência e disponibilidade do principal autor do FreeLing, Lluís Padró. Só com a sua ajuda foi possível resolver vários problemas encontrados durante a implementação deste módulo.

Referências

Atserias, Jordi, Elisabet Comelles, e Aingeru Mayor. 2005. TXALA un analizador libre de dependencias para el castellano. *Procesamiento del Lenguaje Natural*, 35:455–456, Setembro, 2005.

Christ, O. 1994. A Modular and Flexible Architecture for an Integrated Corpus Query System.

Dominus, Mark Jason. 2005. *Higher-Order Perl: Transforming Programs with Programs*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Jenness, Tim e Simon Cozens. 2002. *Extending and Embedding Perl*. Manning Publications, August, 2002.

Jurafsky, D., J.H. Martin, A. Kehler, K. Vander Linden, e N. Ward. 2000. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*, volume 2. Prentice Hall New Jersey.

Padró, Lluís. 2011. Analizadores multilingües en freeling. *Linguamática*, 3(2):13–20, Dezembro, 2011.

Padró, Lluís, Miquel Collado, Samuel Reese, Marina Lloberes, e Irene Castellón. 2010. FreeLing 2.1: five years of open-source language processing tools. La Valletta, Malta, Maio, 2010. ELRA, Proceedings of 7th Language Resources and Evaluation Conference (LREC 2010).

[7] Como já explicamos, neste artigo não pretendemos demonstrar a qualidade do FreeLing, apenas apresentar a interface Perl desenvolvida.

WN-Toolkit: un toolkit per a la creació de WordNets a partir de diccionaris bilingües*

Antoni Oliver
Universitat Oberta de Catalunya
aoliverg@uoc.edu

Resum

En aquest article presentem un conjunt de programes que faciliten la creació de WordNets a partir de diccionaris bilingües mitjançant l'estratègia d'expansió. Els programes estan escrits en Python i són per tant multiplataforma. El seu ús, tot i que no disposen d'una interfície gràfica d'usuari, és molt senzill. Aquests programes s'han fet servir amb èxit en el projecte Know2 per a la creació de les versions 3.0 dels WordNets català i espanyol. Els programes estan publicats sota una llicència GNU-GPL i es poden descarregar lliurement de http://lpg.uoc.edu/wn-toolkit.

Paraules clau

WordNet, estratègia d'expansió

Abstract

This paper presents a set of programs to facilitate the creation of WordNet from bilingual dictionaries following the expand model. The programs are written in Python and are therefore multiplatform. The programs are very easy to use although they don't have a graphical user interface. These programs have been successfully used in the Know2 Project for the creation of Catalan and Spanish WordNet 3.0. The programs are published under the GNU-GPL licence and can be freely downloaded from http://lpg.uoc.edu/wn-toolkit.

Keywords

WordNet, expand model

1 Introducció

WordNet (Fellbaum, 1998) és una base de dades lèxica de l'anglès desenvolupada a la Universitat de Princeton (per aquest motiu, a la resta de l'article anomenarem a aquesta versió PWN - *Princeton WordNet*). En aquesta base de dades les paraules que pertanyen a categories obertes (és a dir, els substantius, verbs, adjectius i adverbis) s'organitzen en conjunts de sinònims denominats *synsets*. Cada *synset* representa un concepte lexicalitzat en anglès, i es connecta amb els altres *synsets* mitjançant una sèrie de relacions semàntiques: hiponímia o relació d'especificitat entre un mot (l'hipònim) i un altre de significat més genèric (hiperònim), l'antonímia o relació entre mots que tenen un significat directament oposat, la meronímia o la relació entre una part i un tot, i la troponímia o implicació lèxica, una relació que es dóna entre verbs i que es pot considerar en certa manera equivalent a la relació d'hiponímia per als substantius. Per exemple, el *synset* del WordNet 3.0 de l'anglès que s'identifica mitjançant un *offset* i una categoria gramatical 02958343-n té assignades diverses variants: *car, auto, automobile, machine i motorcar*. Cada *synset* té assignada una glossa o definició en alguns casos també exemples d'ús. Per al nostre exemple *a motor vehicle with four wheels; usually propelled by an internal combustion engine. he needs a car to get to work.* Aquest *synset* té 31 hipònims, com per exemple 02701002-n (*ambulance*) o 03594945-n (*jeep, landrover*), entre d'altres. També té un hiperònim, 03791235-n (*motor vehicle, automobile vehicle*). D'entre els merònims registrats a WordNet podem posar com a exemple 02685365-n (*airbag*).

1.1 El PWN

El WordNet anglès és lliure i es pot descarregar de la plana web de la Universitat de Princeton[1]. La versió actual és la 3.1 però aquesta versió per ara només es pot consultar *online*. La versió 3.0 es pot descarregar i en aquest article ens centrarem en aquesta versión. A la taula 1 podem observar una comparativa sobre el nombre de *syn-*

*Aquest treball s'ha portat a terme dins del projecte Know2 *Language understanding technologies for multilingual domain-oriented information access* (MICINN, TINN2009-14715-C04-04)

[1] http://wordnet.princeton.edu/

sets per a les versions 1.5, 1.6 y 3.0 del PWN. Com podem veure, el nombre de *synsets* desenvolupats augmenta amb les noves versions.

	1.5	**1.6**	**3.0**
Total	76.705	99.642	118.695
Substantius	51.253	66.025	83.073
Verbs	8.847	12.127	13.845
Adjectius	13.460	17.915	18.156
Adverbis	3.145	3.375	3.621

Taula 1: Comparació del nombre de *synsets* per a tres versions del PWN

1.2 Wordnets per a altres llengües

En diversos projectes s'han desenvolupat wordnets per a altres llengües: EuroWordNet (Vossen, 1998) inicialment per a l'holandès, italià, espanyol i l'ampliació i millora de l'anglès, i en una extensió del projecte per a l'alemany, francès, estonià i txec; el projecte Balkanet (Tufis, Cristea i Stamou, 2004) per al búlgar, grec, romanès, serbi i turc i el projecte RusNet (Azarova et al., 2002) per al rus, entre d'altres projectes similars. A la plana web de la *Global WordNet Association*[2] podem trobar una llista dels WordNets disponibles per a les diferents llengües i amb el seu estat de desenvolupament.

No tots els WordNets que es construeixen es publiquen amb una llicència lliure. A Bond i Kyonghee (2008) podem trobar els WordNets existents per a les diferents llengües amb la llicència associada. Les llengües que disposen de WordNets lliures són: anglès, finès, rus, tailandès, danès, japonès, català, gaèlic, hindi, francès, malai, indonesi, castellà, gallec, basc, àrab i hebreu. De la plana web del projecte *Open Multilingual WordNet* (Bond i Paik, 2012) es poden descarregar un gran nombre de WordNets lliures en un format unificat.

El projecte que ha permès desenvolupar el *toolkit* que presentem en aquest article té com a objectiu desenvolupar els WordNets 3.0 per al català i castellà i distribuir-los sota una llicència lliure.

1.2.1 Estratègies per a la construcció de WordNets

Podem agrupar les estratègies generals per a la construcció de wordnets en dos gran grups: (Vossen, 1998):

- **Estrategia de combinació** (*merge model*): per a cada llengua es genera una ontologia amb els seus propis nivells i relacions. Posteriorment es generen relacions interlingüístiques entre aquesta ontologia i el PWN.
- **Estrategia d'expansió** (*expand model*): es tradueixen les *variants* associades als *synsets* del PWN, fent servir diferents estratègies. Després es verifica si les relacions entre *synsets* donades pel PWN són també vàlides per a la llengua d'arribada.

Vossen (1996) enumera una sèrie d'avantatges i inconvenients per a cada una d'aquestes estratègies. L'estratègia d'expansió és tècnicament més senzilla i garanteix un grau més alt de compatibilitat entre els WordNets de les diferents llengües. Però els WordNets desenvolupats d'aquesta manera estan molt influenciats pel PWN i contindran tots els seus errors i deficiències estructurals. L'estratègia de combinació és més complexa però permet un major aprofitament més directe de les ontologies i tesaurus disponibles.

1.3 Estratègia d'expansió

Com hem comentat, l'estratègia d'expansió consisteix a construir un WordNet per a una llengua determinada traduint les *variants* assignades a cada *synset* del PWN. La manera més evident i emprada és mitjançant l'ús de diccionaris bilingües. La principal dificultat per aplicar l'estratègia d'expansió és la polisèmia. Si totes les *variants* fossin monosèmiques, és a dir, que estiguéssin assignades a un únic *synset* el problema seria simple, ja que només caldria trobar una o mes traduccions per a la *variant* anglesa. Com que la paraula anglesa només tindria un sentit la traducció d'aquesta paraula seria la *variant* correcta en la llengua d'arribada.

Veiem aquest fet amb un exemple. La paraula *aeroplane* està assignada a un únic *synset* (02691156-n) i és, per tant, monosèmica segons el PWN. Si consultem un diccionari anglès-català trobarem dues possibles traduccions: avió i aeroplà. Aquestes dues paraules seran *variants* vàlides per a aquest *synset*. Per una altra banda, la paraula *plane* és polisèmica segons el PWN, ja que està assignada a més d'un *synset* (de fet està assignada als següents *synsets*: 02691156-n, 13861050-n, 13941806-n i 03954731-n). Si traduïm *plane* amb un diccionari anglès-català trobarem les següents possibles traduccions: avió, pla, nivell, garlopa, ribot, plàtan. En aquest cas

[2] `http://www.globalwordnet.org`

no podem assignar totes aquestes paraules com a *variants* vàlides per a tots els *synsets* ni tampoc disposem de suficient informació per a saber a quin *synset* assignar cada una de les *variants*.

A la taula 2 podem observar el nombre de *variants* que tenen assignades un nombre determinat de *synsets*. Les *variants* que tenen assignat un únic *synset* són paraules monosèmiques en anglès (almenys segons PWN). Així, per exemple, el 82.32% de les *variants* del PWN són monosèmiques.

N. synsets	variants	%
1	123.228	82.32
2	15.577	10.41
3	5.027	3.36
4	2.199	1.47
5+	3.659	2.44

Taula 2: Nombre de *variants* que tenen assignades un nombre determinat de *synsets*

Un altre aspecte interessant és observar si aquestes *variants* estan escrites amb la primera lletra en majúscula (i correspondran probablement a un nom propi) i quantes estan escrites amb totes les lletres en minúscules. A la taula 3 podem observar aquests valors.

	variants	%
minúscula	84.714	68.75
majúscula	38.514	31.25

Taula 3: Nombre de *variants* monosèmiques del PWN segons estiguin escrites en minúscules o amb la primera lletra en majúscula

2 Els WordNets per al català i castellà

Els WordNets del castellà (Atserias et al., 1997) i del català (Benítez et al., 1998) es van construir seguint una metodologia d'expansió, ja que es van traduir les *variants* corresponents als *synsets* del PWN. Per als substantius es va fer servir una metodologia basada en diccionaris bilingües. En canvi, els verbs es van desenvolupar d'una manera manual i els adjectius i adverbis no es van desenvolupar en les primeres versions.

A la taula 4 podem observar el nombre de synsets per a les versions 1.6 i 3.0 dels WordNets de l'anglès, català i castellà[3].

[3]La versió 1.6 del castellà no és lliure i en aquesta taula mostrem els valors del fragment lliure distribuït amb l'analitzador Freeling

1.6	Anglès	Català	Castellà
Total	99.645	41.991	21.252
N	66.025	32.236	11.218
V	12.127	5.397	4.994
A	17.915	4.358	5.040
R	3.375	0	0
3.0	**Anglès**	**Català**	**Castellà**
Total	118.695	46.033	38.702
N	83.073	36.460	26.594
V	13.845	5.424	6.251
A	18.156	4.148	5.180
R	3.621	1	677

Taula 4: Nombre de *variants* per a les versions 1.6 i 3.0 de l'anglès, català o castellà

3 Organització de l'article

En aquest article presentem tant el propi *toolkit* com una avaluació dels mètodes que implementa per a la creació dels WordNets 3.0 per al català i castellà. Aquest *toolkit* està format pels programes que hem fet servir en la nostra recerca més una petita documentació per a cada un dels programes. L'objectiu és posar a disposició de la comunitat aquests programes amb l'esperança que sigui útils per a la creació de WordNets per a altres llengües.

La resta de l'article està organitzat de la següent manera. En primer lloc presentarem la tecnologia emprada per a la creació dels programes i els requisits necessaris per a poder-los executar, així com a les instruccions que són comunes per a tots aquests programes. Posteriorment presentarem algunes de les estratègies emprades per a la construcció dels WordNets 3.0 del català i castellà, que són:

- Ús de diccionaris bilingües
- Ús de Babelnet

En la construcció dels WordNets 3.0 per al català i castellà també s'ha fet servir una estratègia basada en l'explotació de corpus paral·lels (Oliver i Climent, 2011; Oliver i Climent, 2012a; Oliver i Climent, 2012b). En aquesta versió del *toolkit* no estan presents els programes necessaris per fer servir aquesta estratègia. Aquests programes es distribuiran en futures versions del *toolkit*.

Per a cada una d'aquestes estratègies es presentarà:

- Una descripció detallada de l'estratègia
- Els programes necessaris per construir WordNets amb aquesta estratègia

- Els resultats de l'avaluació d'aquesta estratègia en la construcció dels WordNets 3.0 per al català i castellà

S'ha dut a terme una avaluació automàtica, consistent en comparar els resultats obtinguts amb les versions preliminars dels WordNets 3.0 del català i castellà. Si una *variant* obtinguda per a un determinat *synset* coincideix amb alguna de les presents en les versions preliminars, aquesta es dóna per correcta. Si no coincideix amb cap de les presents, es considera incorrecta. En cas de no tenir cap *variant* per al *synset* corresponent en les versions preliminars, no s'avalua aquest resultat. Som conscients que els resultats d'aquesta avaluació automàtica poden variar considerablement dels d'una avaluació manual i per aquest motiu en molts casos s'ha portat a terme també una avaluació manual.

4 Aspectes generals sobre el toolkit

El *toolkit* que presentem en aquest article està format per un conjunt de programes escrits en Python. Python és un llenguatge interpretat i l'únic que necessitem per executar aquests programes és disposar del corresponent intèrpret. L'intèrpret es pot descarregar gratuïtament de `http://www.python.org`. Hi ha versions per als sistemes operatius més habituals. Linux i Mac acostumen a tenir instal·lat l'intèrpret de Python per defecte, de manera que si treballeu amb aquests sistemes operatius no necessitareu instal·lar res al vostre ordinador. Els programes que presentem no disposen d'interfície gràfica d'usuari i funcionen sota línia de comandes (*Terminal* en Linux i Mac i *Símbol de sistema* en Windows).

Els programes s'executen donant una sèrie de paràmetres que variarà segons el programa. Per saber els paràmetres que cal donar podeu fer:

```
python nomprograma.py -h
```

on *nomprograma.py* és el nom del programa que volem executar.

El Toolkit es pot descarregar de `http://lpg.uoc.edu/wn-toolkit`.

5 Ús de diccionaris bilingües

5.1 Descripció de l'estratègia

Amb aquesta primera estratègia obtenim *variants* únicament per als *synsets* les *variants* en anglès dels quals són monosèmiques. És a dir, traduïm mitjançant diferents tipus de diccionaris (generals, enciclopèdics i terminològics) paraules angleses monosèmiques (assignades a un únic *synset*) i assignem aquest *synset* a la corresponent paraula o paraules de la llengua d'arribada donades pel diccionari.

5.2 Programes

Per fer ús d'aquesta estratègia hem de fer servir diversos programes:

- **createmonosemicwordlist.py**: per crear les llistes de paraules monosèmiques del PWN anglès. Alternativament, podem fer servir directament les llistes de paraules monosèmiques que es distribueixen amb el *toolkit* corresponents a la versió 3.0.
- **wndictionary.py**: a partir d'una llista de paraules monosèmiques del PWN anglès i d'un diccionari bilingüe proporciona una llista de *synsets* amb les seves corresponents *variants* en la llengua d'arribada.
- **apertium2bildic.py**: crea un diccionari bilingüe adequat per fer-lo servir amb el programa wndictionary.py a partir dels diccionaris de transferència del sistema de traducció automàtica Apertium (Forcada, Tyers i Ramírez-Sánchez, 2009).
- **dacco2bildic.py**: crea un diccionari bilingüe adequat per fer-lo servir amb el programa wndictionary.py a partir del diccionari de lliure distribució anglès-català Dacco[4]
- **TO2bildic.py**: crea un diccionari bilingüe adequat per fer-lo servir amb el programa wndictionary.py a partir dels glossaris terminològics Terminologia Oberta del TermCat[5].
- **wiktionary2bildic.py**: crea un diccionari bilingüe adequat per fer-lo servir amb el programa wndictionary.py a partir dels fitxers dump xml del Wiktionary[6]
- **wikipedia2bildic.py**: crea un diccionari enciclopèdic bilingüe adequat per fer-lo servir amb el programa wndictionary.py a partir del fitxers dump xml de la Vikipèdia[7]
- **combinedictionary.py**: aquest programa permet combinar diversos diccionaris, de manera que es crea un únic diccionari que conté la informació de tots ells, sense reduplicar la informació comuna.

[4] `http://www.catalandictionary.org/`
[5] `http://.termcat.org`
[6] `www.wiktionary.org`
[7] `www.wikipedia.org`

En els següents subapartats expliquem amb més detall cada un d'aquests programes.

5.2.1 createmonosemicwordlist.py

Aquest programa extreu tres llistes de *variants* monosèmiques del PWN. Per poder executar el programa primer hem de descarregar els arxius corresponents a les bases de dades del WordNet des de `http://wordnetcode.princeton.edu/3.0/WNdb-3.0.tar.gz` (o l'arxiu corresponent a la versió desitjada). El programa pren com a paràmetres el directori on es troben els arxius de WordNet i el prefix que volem que tinguin els arxius de sortida. Es creen tres arxius: un que conté totes les variants monosèmiques, un que conté les que estan escrites en minúscules i un altre que conté les que estan escrites amb la primera lletra en majúscula. Per executar el programa podem fer:

```
python createmonosemicwordlist.py -d ./dict
-p pwnmonosemic
```

Els arxius de sortida contenen l'offset i categoria gramatical i la variant monosèmica separats per una tabulador, com al següent exemple:

```
02691156n aeroplane
02691156n airplane
```

Amb el *toolkit* es distribueixen els fitxers corresponents a les variants monosèmiques de la versió 3.0.

5.2.2 wndictionary.py

Amb aquest programa podem obtenir las variants en la llengua d'arribada a partir d'una llista de variants angleses monosèmiques obtingudes amb createmonosemicwordlist.py. El programa demana el fitxer de diccionari que volem fer servir, el fitxer d'entrada amb les variants monosèmiques i el nom del fitxer de sortida. Per executar el programa podem fer:

```
python wndictionary.py -d diccionari.txt
-i pwnmonosemic.txt -o wordnetcreat.txt
```

La sortida té la següent forma:

```
02691156n avión
02691156n aeroplano
```

5.2.3 apertium2bildic.py

Aquest programa permet crear un diccionari bilingüe a partir dels diccionaris de transferència del sistema de traducció automàtica Apertium[8]. Aquest sistema de traducció automàtica es distribueix sota llicència lliure i es poden descarregar les dades lingüístiques des de `http://sourceforge.net/projects/apertium/files/`. Triarem el parell de llengües desitjat i descarregarem l'arxiu corresponent a la darrera versió. Un cop descomprimit l'arxiu seleccionarem l'arxiu corresponent al diccionari de transferència (que s'anomena per l'anglès-català: apertium-en-ca.en-ca.dix).

```
python apertium2bildic.py -a
apertium-en-ca.en-ca.dix -o
diccionari-en-ca.txt
```

En els casos en què l'anglès no estigui com a primera llengua del diccionari, caldrà fer servir l'opció -r, per canviar l'ordre de les entrades i tenir l'anglès en primer lloc. La sortida és un fitxer de text tabulat amb paraula en anglès, categoria gramatical i paraula en la llengua d'arribada, com al següent exemple:

```
caution n amonestació
cautious a cautelós
cautiously r amb cautela
```

5.2.4 dacco2bildic.py

Dacco és un diccionari lliure col·laboratiu anglès-català i català-anglès. Es poden descarregar els arxius de `http://sourceforge.net/projects/dacco/`. Descarreguem l'arxiu dacco-0.9.zip (o el corresponent a la darrera versió disponible) i el descomprimim. Podem executar el programa donant com a paràmetres el directori on es troben els diccionaris anglès-català i el fitxer de sortida, per exemple:

```
python dacco2bildic.py
./Dacco-0.9/dictionaries/engcat
-o diccionaridacco.txt
```

La sortida és un fitxer de text tabulat amb paraula en anglès, categoria gramatical i paraula en català, com al següent exemple:

```
last a últim:darrer
last v durar
last name n cognom
```

5.2.5 TO2bildic.py

Aquest programa transforma un o més glossaris terminològics de Terminologia Oberta del TermCat en un diccionari en format per ser utilitzat amb el programa wndictionary.py. Els glossaris de Terminologia Oberta del TermCat es poden descarregar de `http://www.termcat.cat/productes/toberta.htm`. Per fer funcionar aquest programa hem de descarregar almenys un dels glossaris (es pot treballar amb més d'un alhora i fins i tot amb tots). Posem tots els glossaris en un directori i els descomprimim. Al programa passarem com a paràmetres el glossari a tractar o el directori que conté tots els glossaris que volem tractar i el fitxer de sortida. Per exemple, si volem tractar tots els glossaris que estan al directori /home/TO i posar el resultat en un arxiu anomenant TO.txt hem de fer:

[8] `http://www.apertium.org`

```
python TO2bildic.py /home/TO -O TO.txt
```

Donat que els glossaris de Terminologia Oberta no especifiquen la categoria gramatical, assignem "n"(substantiu) a totes les entrades, ja que la immensa majoria de les entrades corresponen a aquesta categoria gramatical. La sortida és un fitxer de text tabulat amb paraula en anglès, categoria gramatical i paraula en la llengua d'arribada, com al següent exemple:

```
fifth disease n eritema infecciós
fight n baralla
fighting n baralla
```

5.2.6 wiktionary2bildic.py

Aquest programa serveix per a generar un diccionari bilingüe a partir dels fitxers *dump xml* de Wiktionary[9]. Wiktionary és un projecte col·laboratiu per a la creació de diccionaris en moltes llengües, amb enllaços interlingüístics. És possible descarregar els fitxers *dump xml* de `http://dumps.wikimedia.org/`. Com que per a la creació de WordNets ens interessen diccionaris d'anglès a la llengua d'arribada, descarregarem els fitxers corresponents al Wiktionary anglès, des de `http://dumps.wikimedia.org/enwiktionary/`. És recomanable fer servir sempre la darrera versió disponible. Cal tenir en compte que aquests fitxers són molt grans i que la descàrrega pot durar molt de temps. També cal preveure tenir espai en disc disponible.

```
python wiktionary2bildic.py
enwiktionary-latest-pages-articles.xml
-l Catalan -o wiktionary-eng-cat.txt
```

Cal tenir en compte que les llengües s'especifiquen amb el nom complet en anglès (Catalan, Spanish, French...) A continuació observem una mostra del resultat:

```
listen v escoltar
literally r literalment
literary a literari
```

5.2.7 wikipedia2bildic.py

Aquest programa és semblant al wiktionary2bildic.py però crea un diccionari a partir dels fitxers *dump xml* de Wikipedia[10]. Els diccionaris que es creïn seran uns diccionaris de caire enciclopèdic donada la pròpia naturalesa de la Vikipèdia. És possible descarregar els fitxers *dump xml* de `http://dumps.wikimedia.org/`. Com que per a la creació de WordNets ens interessen diccionaris d'anglès a la llengua d'arribada, descarregarem els fitxers corresponents a la Vikipèdia anglesa, des de `http://dumps.wikimedia.org/enwiki/`. És recomanable fer servir sempre la darrera versió disponible. Cal tenir en compte que aquests fitxers són molt grans i que la descàrrega pot durar molt de temps. També cal preveure tenir espai en disc disponible.

Per executar el programa simplement s'ha de fer:

```
python wikipedia2bildic.py
enwiki-latest-pages-articles.xml
-l ca -o wikipedia-eng-cat.txt
```

Cal tenir en compte que les llengües s'especifiquen amb els codi ISO de dues lletres (ca, es, fr...) A continuació observem una mostra del resultat:

```
Gregor Mendel n Gregor Mendel
Grammar n Gramàtica
Gigabyte n Gigabyte
Galaxy groups and clusters n
Cúmul de galàxies
```

A la Vikipèdia no disposem d'informació sobre la categoria gramatical, però la immensa majoria seran substantius i per aquest motiu assignem la categoria n a totes les entrades.

5.2.8 combinedictionary.py

Aquest programa permet combinar dos o més diccionaris en un de sol. Té cura de no repetir ni entrades ni accepcions. Els paràmetres que cal donar són dos o més diccionaris d'entrada seguit del nom del diccionari de sortida. El programa verifica que el fitxer de sortida no existeixi, per evitar sobreescriure un fitxer existent.

```
python combinedictionary.py dict1.txt
dict2.txt dict3.txt dictsortida.txt
```

5.3 Avaluació

5.3.1 Ús de diccionaris bilingües generals

En aquest experiment hem generat els WordNets a partir de diccionaris obtinguts a partir dels diccionaris de transferència d'Apertium i del Wiktionary. A la taula 5 podem observar el nombre d'entrades de cada un d'aquests diccionaris, així com el nombre d'entrades del diccionari resultant de combinar les dues fonts.

Diccionari	**eng-spa**	**eng-cat**
Apertium	20.366	29.154
Wiktionary	23.196	7.393
Total	34.600	32.921

Taula 5: Nombre d'entrades dels diccionaris bilingües generals

Per al castellà podem obtenir un total de 12.676 *variants* de las que 7.401 són correctes, 2.997 incorrectes (segons l'avaluació automàtica) i no podem avaluar 2.278. La precisió per al castellà, segons l'avaluació

[9] `http://www.wiktionary.org`
[10] `http://www.wikipedia.org`

automàtica és del 71.2%. S'han revisat manualment tots els resultats considerats incorrectes per l'avaluació automàtica. Això ens ha permès calcular una nova precisió, que ara puja fins el 93.95%.

Per al català obtenim un total de 8.335 *variants*, de les que 4.223 són correctes, 1.083 incorrectes (segons l'avaluació automàtica) y no podem avaluar 3.029. La precisió per al català, segons l'avaluació automàtica, és del 79.6%. De la mateixa manera que per al castellà, hem revisat els resultats incorrectes i hem pogut calcular una nova precisió que ara puja fins el 96.36%.

5.3.2 Ús de diccionaris enciclopèdics

En aquest experiment s'ha fet servir un diccionari enciclopèdic per a traduir les *variants* angleses monosèmiques escrites amb la primera lletra en majúscula. Aquestes constitueixen el 31.15% de les *variants* monosèmiques, com es pot veure a la taula 3. D'aquestes, la immensa majoria (99.17%) són substantius.

S'ha creat un diccionari enciclopèdic bilingüe anglès castellà de 59.659 entrades i anglès-català de 22.205 entrades a partir de la Vikipèdia anglesa fent servir el programa wikipedia2bildic.py.

Per al castellà podem obtenir un total de 10.356 *variants* de les que 4.722 són correctes. 1.916 incorrectes (segons l'avaluació automàtica) i no podem avaluar 3.718. La precisió per al castellà, segons aquesta avaluació automàtica, és del 71.1%. Si revisem els casos donats per incorrectes i recalculem la precisió, aquesta augmenta fins el 89.74%.

Per al català obtenim un total de 7.083 *variants*, de les que 2.642 són correctes, 1.278 incorrectes (segons l'avaluació automàtica) i no podem avaluar 3.163. La precisió per al català, segons l'avaluació automàtica, és del 67.4%. Després de la revisió manual dels classificats com a incorrectes, la precisió augmenta fins el 90.94%.

5.3.3 Ús de diccionaris terminològics

En aquest experiment hem fet servir un conjunt de diccionaris terminològics per a traduir les *variants* angleses monosèmiques, tant les que estan escrites en minúscules com les que tenen la primera lletra en majúscula. Hem obtingut un diccionari terminològic fent servir el programa TO2bildic.py a partir de tots els glossaris terminològics de Terminologia Oberta del TermCat. D'aquesta manera hem confeccionat un diccionari terminològic anglès-castellà de 46.761 entrades i un anglès-català de 46.653 entrades.

Per al castellà obtenim un total de 10.456 *variants*, de les que 4.180 són correctes, 3.346 incorrectes (segons l'avaluació automàtica) i no podem avaluar 2.930. La precisió per al castellà, segons l'avaluació automàtica, és del 55.5%. Aquest resultat és molt baix i decidim revisar manualment tant les avaluades automàticament com a incorrectes, com les no avaluades. Moltes d'aquestes eren en realitat correctes i la nova precisió augmenta fins el 98.57%.

Per al català podem obtenir un total de 9.890 *variants* de les que 3.007 són correctes, 2.614 incorrectes (segons l'avaluació automàtica) i no podem avaluar 4.269. La precisió per al català calculada de manera automàtica és del 53.6% però si revisem manualment els resultats la precisió augmenta dins el 98.36%.

6 Babelnet

6.1 Descripció

Babelnet (Navigli i Ponzetto, 2010) és una xarxa semàntica de grans dimensions que s'ha creat combinant el coneixement lexicogràfic de WordNet amb el coneixement enciclopèdic de la Vikipèdia. D'aquesta manera Babelnet ofereix una relació entre els *synsets* de WordNet i les entrades de la Vikipèdia. Aquesta relació s'ha obtingut tant per a entrades amb títols monosèmics com polisèmics. Així, aprofitant aquest recurs podrem obtenir *variants* en altres llengües independentment si la *variant* anglesa associada al *synset* és monosèmica com si és polisèmica.

Per poder relacionar les dues fonts els autors prenen de WordNet tots els possibles sentits d'una determinada paraula i totes les relacions semàntiques dels *synsets*. De la Vikipèdia prenen totes les entrades i les relacions donades pels enllaços d'hipertext de les pàgines. Aquestes relacions poden ser de diferents tipus i no estan especificades des del punt de vista semàntic. Per establir un *mapping* entre els dos recursos fan servir els anomenats *contextos de desambiguació*. Aquests contextos, per als articles de la Vikipèdia estan formats per les etiquetes de sentit que tenen algunes entrades, els enllaços d'hipertext i les categories. En el cas de WordNet aquests contextos estan formats per tots els sinònims, hiperònims i hipònims, els lemes de les categories obertes de la glossa o definició i les *variants* associades als *synsets* germans, és a dir, els que tenen un hiperònim directe comú. Per establir els *mappings* apliquen els següents criteris:

- Per a totes les planes de la Vikipèdia que tinguin un títol monosèmic tant per la Vikipèdia com per WordNet, s'enllaça directament la plana amb el *synset*.
- Per a la resta de planes es calcula la intersecció dels contextos de desambigüació per a tots els sentits de la Vikipèdia i WordNet.

Aprofitant els enllaços interlingüístics de la Vikipèdia, aquesta relació es pot establir per a totes les llengües que disposin de l'entrada corresponent.

La versió 3.0 de Babelnet oferia un arxiu anomentat babel-to-wordnet-3.0.txt que tenia el següent aspecte:

```
Adobe_brick adobe_brick%1:06:00:: 02681392n
Fuselage fuselage%1:06:00:: 03408054n
Hearse hearse%1:06:00:: 03506880n
Merida_(Yucatan) merida%1:15:00:: 08740367n
```

és a dir, relacionava el títol d'una entrada de la Vikipèdia anglesa amb una *variant* i un *synset* del

Princeton WordNet 3.0. Els experiments que hem portat a terme s'han fet amb aquesta versió i els resultats que oferim a l'apartat d'Avaluació són els obtinguts amb aquesta versió.

La versió actual disponible difereix en el format i el continguts (Navigli i Ponzetto, 2012). La distribució inclou els següents arxius:

- BabelNet API: és una API escrita en Java per accedir a la informació de Babelnet.
- BabelNet precompiled index: es tracta dels índexs precompil·lats
- BabelNet glosses: Aquest és el que farem servir per extreure informació, ja que conté la relació entre els *synsets* de BabelNet i WordNet i les entrades de la Vikipèdia.

Mirem més a fons el contingut del fitxer BabelNet glosses:

```
bn:00001439n
CA WIKI Almirall Almiral l'és el grau militar,
o part del nom del ranc , amb que es coneixen
els caps d'una flota o marina de guerra.
ES WIKI Almirante Almirante es un grado
militar de la marina de guerra que equivale
al de general en otros cuerpos del ejército.
IT WIKI Ammiraglio Il grado di Ammiraglio
è più alto nella gerarchia delle odierne
marine militari .
DE WIKI Admiral Admiral ist der höchste
militärische Dienstgrad in der Marine,
entsprechend dem General des Heeres und der
Luftwaffe.
EN WIKIWN 09771204n the supreme commander
of a fleet; ranks above a vice admiral and
below a fleet admiral
EN WIKI Admiral  Admiral is the rank , or
part of the name of the ranks , of the
highest naval officers.
```

Si agafem la informació de la línia EN WIKIWN, que és el *synset* de WordNet (09771204n), podem saber directament que una possible *variant* en català és *Almirall*. Aquesta informació és directament deduïble des d'aquest fitxer, ja que inclou el català. Si estem construint un WordNet per a una altra llengua, podríem consultar els enllaços interlingüístics de la Vikipèdia anglesa corresponent a l'entrada *Admiral*. Per exemple, podríem deduir que en holandès és *Admiraal*.

El recurs es pot descarregar de la plana web `http://lcl.uniroma1.it/babelnet/` i també ofereix una interfície de consulta.

6.2 Programes

El programa babel2wordnet.py pren com a paràmetres l'arxiu de glosses de Babelnet i, de manera opcional, un diccionari creat per a la llengua d'arribada desitjada mitjançant el programa wikipedia2bildic.py. Si volem generar un WordNet per alguna de les llengües incloses al fitxer de glosses de Babelnet no serà imprescindible indicar un diccionari; si no en donem cap, simplement extraurà la informació continguda al BabelNet. Si proporcionem un diccionari completarà la informació inclosa en el BabelNet. Per a llengües no incloses a Babelnet, es del tot imprescindible proporcionar un diccionari.

El programa, doncs, funciona de la següent manera:

```
python babel2wordnet.py babel-glosses
-l ru -d diccionari_wikipedia-rus.txt
-o babelwordnet-rus.txt
```

El programa també pot intentar unificar les majúscules i minúscules a partir de la informació continguda en el propi WordNet. Per aconseguir això, s'ha de donar el directori on es troben els arxius de WordNet mitjançant el paràmetre -w.

```
python babel2wordnet.py babel-glosses
-l ru -d diccionari_wikipedia-rus.txt
-o babelwordnet-rus.txt
-w /home/usuari/WordNet30
```

Aquesta opció s'ha de fer servir únicament per a aquelles llengües on la capitalització o no de les paraules segueixi el mateix patró que l'anglès, és a dir, noms propis escrits en majúscules.

La sortida que ens proporcionarà serà com la següent:

```
09771204n almirall
08784104n Eólida
03745571n menhir
12154426n Pandanàcia
12960211n Ophioglossum
00149895n soldadura per punts
```

6.3 Avaluació

Aquesta avaluació s'ha dut a terme amb la versió antiga de BabelNet, és a dir, fent servir el fitxer babelto-wordnet-3.0.txt.

Per al castellà obtenim un total de 26.209 *variants*, de las que 14.614 són correctes, 5.065 incorrectes (segons l'avaluació automàtica) i no podem avaluar automàticament 6.530. La precisió per al castellà, segons aquesta avaluació automàtica, és del 74.3%. Revisem manualment tant les avaluades automàticament com a incorrectes, com les no avaluades. Un cop portada a terme aquesta avaluació manual podem calcular un nou valor de precisió, que és ara del 81.02%.

Per al català podem obtenir un total de 18.366 *variants* de les que 9.044 són correctes, 3.548 incorrectes (segons l'avaluació automàtica) i no podem avaluar 5.774. La precisió per al català, segons l'avaluació automàtica, és del 61%. Un cop revisades tant les avaluades automàticament com a incorrectes com les no avaluades podem calcular una nova precisió que és ara del 80.91%.

7 Conclusions

En aquest article hem presentat el WN-Toolkit per a la creació de WordNets seguint l'estratègia d'expansió mitjançant l'ús de diccionaris bilingües. Hem afegit també programes per crear WordNets a partir de Babelnet. Tots aquests algorismes s'han fet servir amb èxit per a la creació dels WordNet 3.0 del català i castellà.

A l'article presentem també els resultats de l'avaluació d'aquesta metodologia. Les estratègies basades en diccionaris només poden obtenir *variants* per a *synsets* que tinguin assignades *variants* monosèmiques. La metodologia basada en Babelnet no presenta aquesta restricció

Per a la creació dels WordNets del català i castellà també es van fer servir mètodes basats en corpus paral·lels(Oliver i Climent, 2011; Oliver i Climent, 2012a; Oliver i Climent, 2012b). Aquestes metodologies basades en corpus paral·lels no presenten la restricció de les metodologies basades en diccionaris pel que fa a la monosèmia de les *variants* angleses assignades als *synsets*. En una propera versió d'aquest Toolkit afegirem els programes necessaris per replicar aquesta metodologia.

Bibliografía

Atserias, J., S. Climent, X. Farreres, G. Rigau, i H. Rodriguez. 1997. Combining multiple methods for the automatic construction of multi-lingual WordNets. Em *Recent Advances in Natural Language Processing II. Selected papers from RANLP*, volume 97, pp. 327–338.

Azarova, I., O. Mitrofanova, A. Sinopalnikova, M. Yavorskaya, i I. Oparin. 2002. Russnet: Building a lexical database for the Russian language. Em *Workshop on WordNet Structures and Standarisation, and how these affect WordNet Application and Evaluation*, pp. 60–64, Las Palmas de Gran Canaria (Spain).

Benítez, Laura, Sergi Cervell, Gerard Escudero, Mònica López, German Rigau, i Mariona Taulé. 1998. Methods and tools for building the catalan WordNet. Em *In Proceedings of the ELRA Workshop on Language Resources for European Minority Languages*.

Bond, F. i P. Kyonghee. 2008. A survey of wordnets and their licenses. Em *Proceedings of the 6th International Global WordNet Conference, Matsue (Japan)*, pp. 64–71.

Bond, F. i K. Paik. 2012. A survey of WordNets and their licenses. Em *Proceedings of the 6th Global WordNet Conference (GWC 2012)*, volume 8, pp. 5, Matsue (Japan).

Fellbaum, C. 1998. *WordNet: An electronic lexical database*. The MIT press.

Forcada, M. L, F. M Tyers, i G. Ramírez-Sánchez. 2009. The apertium machine translation platform: five years on. Em *Proceedings of the First International Workshop on Free/Open-Source Rule-Based Machine Translation*, pp. 3–10.

Navigli, R. i S. Ponzetto. 2012. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence, Elsevier*.

Navigli, Roberto i Simone Paolo Ponzetto. 2010. BabelNet: building a very large multilingual semantic network. Em *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pp. 216–225, Stroudsburg, PA, USA. Association for Computational Linguistics. ACM ID: 1858704.

Oliver, A. i S. Climent. 2011. Construcción de los wordnets 3.0 para castellano y catalán mediante traducción automática de corpus anotados semánticamente. Em *Proceedings of the 27th Conference of the SEPLN, Huelva Spain*.

Oliver, A. i S. Climent. 2012a. Building wordnets by machine translation of sense tagged corpora. Em *Proceedings of the Global WordNet Conference, Matsue, Japan*.

Oliver, A. i S. Climent. 2012b. Parallel corpora for wordnet construction. Em *Proceedings of the 13th International Conference on Intelligent Text Processing and Computational Linguistics (Cicling 2012). New Delhi (India)*.

Tufis, D., D. Cristea, i S. Stamou. 2004. BalkaNet: aims, methods, results and perspectives: a general overview. *Science and Technology*, 7(1-2):9–43.

Vossen, P. 1996. Right or wrong. combining lexical resources in the EuroWordNet project. Em *Proceedings of Euralex-96*, pp. 715–728, Goetheborg.

Vossen, P. 1998. Introduction to Eurowordnet. *Computers and the Humanities*, 32(2):73–89.

Chamada de Artigos

A revista Linguamática pretende colmatar uma lacuna na comunidade de processamento de linguagem natural para as línguas ibéricas. Deste modo, serão publicados artigos que visem o processamento de alguma destas línguas.

A Linguamática é uma revista completamente aberta. Os artigos serão publicados de forma electrónica e disponibilizados abertamente para toda a comunidade científica sob licença *Creative Commons*.

Tópicos de interesse:

- Morfologia, sintaxe e semântica computacional
- Tradução automática e ferramentas de auxílio à tradução
- Terminologia e lexicografia computacional
- Síntese e reconhecimento de fala
- Recolha de informação
- Resposta automática a perguntas
- Linguística com corpora
- Bibliotecas digitais
- Avaliação de sistemas de processamento de linguagem natural
- Ferramentas e recursos públicos ou partilháveis
- Serviços linguísticos na rede
- Ontologias e representação do conhecimento
- Métodos estatísticos aplicados à língua
- Ferramentas de apoio ao ensino das línguas

Os artigos devem ser enviados em PDF através do sistema electrónico da revista. Embora o número de páginas dos artigos seja flexível sugere-se que não excedam 20 páginas. Os artigos devem ser devidamente identificados. Do mesmo modo, os comentários dos membros do comité científico serão devidamente assinados.

Em relação à língua usada para a escrita do artigo, sugere-se o uso de português, galego, castelhano, basco ou catalão.

Os artigos devem seguir o formato gráfico da revista. Existem modelos LaTeX, Microsoft Word e OpenOffice.org na página da Linguamática.

Datas Importantes

- Envio de artigos até: 15 de Abril de 2013
- Resultados da selecção até: 15 de Maio de 2013
- Versão final até: 31 de Maio de 2013
- Publicação da revista: Junho de 2013

Qualquer questão deve ser endereçada a: `editores@linguamatica.com`

Petición de Artigos

A revista Linguamática pretende cubrir unha lagoa na comunidade de procesamento de linguaxe natural para as linguas ibéricas. Deste xeito, han ser publicados artigos que traten o procesamento de calquera destas linguas.

Linguamática é unha revista completamente aberta. Os artigos publicaranse de forma electrónica e estarán ao libre dispor de toda a comunidade científica con licenza *Creative Commons*.

Temas de interese:

- Morfoloxía, sintaxe e semántica computacional
- Tradución automática e ferramentas de axuda á tradución
- Terminoloxía e lexicografía computacional
- Síntese e recoñecemento de fala
- Extracción de información
- Resposta automática a preguntas
- Lingüística de corpus
- Bibliotecas dixitais
- Avaliación de sistemas de procesamento de linguaxe natural
- Ferramentas e recursos públicos ou cooperativos
- Servizos lingüísticos na rede
- Ontoloxías e representación do coñecemento
- Métodos estatísticos aplicados á lingua
- Ferramentas de apoio ao ensino das linguas

Os artigos deben de enviarse en PDF mediante o sistema electrónico da revista. Aínda que o número de páxinas dos artigos sexa flexíbel suxírese que non excedan as 20 páxinas. Os artigos teñen que identificarse debidamente. Do mesmo modo, os comentarios dos membros do comité científico serán debidamente asinados.

En relación á lingua usada para a escrita do artigo, suxírese o uso de portugués, galego, castelán, éuscaro ou catalán.

Os artigos teñen que seguir o formato gráfico da revista. Existen modelos LaTeX, Microsoft Word e OpenOffice.org na páxina de Linguamática.

Datas Importantes

- Envío de artigos até: 15 de abril de 2013
- Resultados da selección: 15 de maio de 2013
- Versión final: 31 de maio de 2013
- Publicación da revista: xuño de 2013

Para calquera cuestión, pode dirixirse a: `editores@linguamatica.com`

Petición de Artículos

La revista Linguamática pretende cubrir una laguna en la comunidad de procesamiento del lenguaje natural para las lenguas ibéricas. Con este fin, se publicarán artículos que traten el procesamiento de cualquiera de estas lenguas.

Linguamática es una revista completamente abierta. Los artículos se publicarán de forma electrónica y se pondrán a libre disposición de toda la comunidad científica con licencia *Creative Commons*.

Temas de interés:

- Morfología, sintaxis y semántica computacional
- Traducción automática y herramientas de ayuda a la traducción
- Terminología y lexicografía computacional
- Síntesis y reconocimiento del habla
- Extracción de información
- Respuesta automática a preguntas
- Lingüística de corpus
- Bibliotecas digitales
- Evaluación de sistemas de procesamiento del linguage natural
- Herramientas y recursos públicos o cooperativos
- Servicios lingüísticos en la red
- Ontologías y representación del conocimiento
- Métodos estadísticos aplicados a la lengua
- Herramientas de apoyo para la enseñanza de lenguas

Los artículos tienen que enviarse en PDF mediante el sistema electrónico de la revista. Aunque el número de páginas de los artículos sea flexible, se sugiere que no excedan las 20 páginas. Los artículos tienen que identificarse debidamente. Del mismo modo, los comentarios de los miembros del comité científico serán debidamente firmados.

En relación a la lengua usada para la escritura del artículo, se sugiere el uso del portugués, gallego, castellano, vasco o catalán.

Los artículos tienen que seguir el formato gráfico de la revista. Existen modelos LaTeX, Microsoft Word y OpenOffice.org en la página de Linguamática.

Fechas Importantes

- Envío de artículos hasta: 15 de abril de 2013
- Resultados de la selección: 15 de mayo de 2013
- Versión final: 31 de mayo de 2013
- Publicación de la revista: junio de 2013

Para cualquier cuestión, puede dirigirse a: `editores@linguamatica.com`

Petició d'articles

La revista Linguamática pretén cobrir una llacuna en la comunitat del processament de llenguatge natural per a les llengües ibèriques. Així, es publicaran articles que tractin el processament de qualsevol d'aquestes llengües.

Linguamática és una revista completament oberta. Els articles es publicaran de forma electrònica i es distribuiran lliurement per a tota la comunitat científica amb llicència *Creative Commons*.

Temes d'interès:

- Morfologia, sintaxi i semàntica computacional
- Traducció automàtica i eines d'ajuda a la traducció
- Terminologia i lexicografia computacional
- Síntesi i reconeixement de parla
- Extracció d'informació
- Resposta automàtica a preguntes
- Lingüística de corpus
- Biblioteques digitals
- Evaluació de sistemes de processament del llenguatge natural
- Eines i recursos lingüístics públics o cooperatius
- Serveis lingüístics en xarxa
- Ontologies i representació del coneixement
- Mètodes estadístics aplicats a la llengua
- Eines d'ajut per a l'ensenyament de llengües

Els articles s'han d'enviar en PDF mitjançant el sistema electrònic de la revista. Tot i que el nombre de pàgines dels articles sigui flexible es suggereix que no ultrapassin les 20 pàgines. Els articles s'han d'identificar degudament. Igualement, els comentaris dels membres del comitè científic seràn degudament signats.

En relació a la llengua usada per l'escriptura de l'article, es suggereix l'ús del portuguès, gallec, castellà, basc o català.

Els articles han de seguir el format gràfic de la revista. Es poden trobar models LaTeX, Microsoft Word i OpenOffice.org a la pàgina de Linguamática.

Dades Importants

- Enviament d'articles fins a: 15 d'abril de 2013
- Resultats de la selecció: 15 de maig de 2013
- Versió final: 31 de maig de 2013
- Publicació de la revista: juny de 2013

Per a qualsevol qüestió, pot adreçar-se a: `editores@linguamatica.com`

Artilulu eskaera

Iberiar penintsulako hizkuntzei dagokienean, hizkuntza naturalen prozedura komunitatean dagoen hutsunea betetzea litzateke Linguamática izeneko aldizkariaren helburu nagusiena. Helburu nagusi hau buru, aurretik aipaturiko edozein hizkuntzen prozedura landuko duten artikuluak argitaratuko dira.

Linguamática aldizkaria irekia da oso. Artikuluak elektronikoki argitaratuko dira, eta komunitate zientefikoaren eskura egongo dira honako lizentziarekin; *Creative Commons*.

Gai interesgarriak:

- Morfologia, sintaxia eta semantika konputazionala.
- Itzulpen automatikoa eta itzulpengintzarako lagungarriak diren tresnak.
- Terminologia eta lexikologia konputazionala.
- Mintzamenaren sintesia eta ikuskapena.
- Informazio ateratzea.
- Galderen erantzun automatikoa.
- Corpus-aren linguistika.
- Liburutegi digitalak.
- Hizkuntza naturalaren prozedura sistemaren ebaluaketa.
- Tresna eta baliabide publikoak edo kooperatiboak.
- Zerbitzu linguistikoak sarean.
- Ezagutzaren ontologia eta adierazpideak.
- Hizkuntzean oinarrituriko metodo estatistikoak.
- Hizkuntzen irakaskuntzarako laguntza tresnak.

Arikuluak PDF formatoan eta aldizkariaren sitema elektronikoaren bidez bidali behar dira. Orri kopurua malgua den arren, 20 orri baino gehiago ez idaztea komeni da. Artikuluak behar bezala identifikatu behar dira. Era berean, zientzi batzordeko kideen iruzkinak ere sinaturik egon beharko dira.

Artikulua idazterako garaian, erabilitako hizkuntzari dagokionean, honako kizkuntza hauek erabili daiztezke; portugesa, galiziera, gaztelania, euskara, eta katalana.

Artikuluek, aldizkariaren formato grafikoa jarraitu behar dute. "Linguamática" orrian LaTeX, Microsoft Word eta OpenOffice.org ereduak aurki ditzakegu.

Data garratzitsuak:

- Arikuluak bidali ahal izateko epea: 2013eko apirilak 15.
- Hautapenaren emaitzak: 2013eko maiatzak 15.
- Azken itzulpena: 2013eko maiatzak 31.
- Aldizkariaren argitarapena: 2013eko ekainean.

Edozein zalantza argitzeko, hona hemen helbide hau: `editores@linguamatica.com`.

Volume 4, Número 2- Dezembro 2012

linguaMATICA

ISSN: 1647-0818