DESAMBIGUADOR DE ETIQUETAGEM DIRIGIDO POR REGRAS LINGUÍSTICAS

CAROLINE HAGÈGE, ANTÓNIO MEIRELES, CARLA DIOGO
FERNANDO LEITE, NAZARÉ BARÃO, PAULO COTOVIO
(ILTEC)

1. Introdução

No momento em que a maioria dos documentos de grandes dimensões e de elevada difusão é elaborada recorrendo ao uso de ferramentas de processamento automático de texto, torna-se necessário o desenvolvimento de ferramentas de correcção sintáctica para o português que permitam detectar e corrigir os erros mais comuns que o utilizador da língua comete. Neste quadro, o projecto RECTIS visa desenvolver um protótipo de um Verificador Sintáctico voltado para as necessidades do utilizador comum do português europeu.

Este projecto pressupõe a construção de três módulos: um Analisador morfossintáctico, um Desambiguador de etiquetagem e o Verificador Sintáctico. O módulo de desambiguação, objecto desta comunicação, tem a dupla função de desambiguar a etiquetagem fornecida pelo Etiquetador e realizar um esboço de análise gramatical - pré-sintaxe.

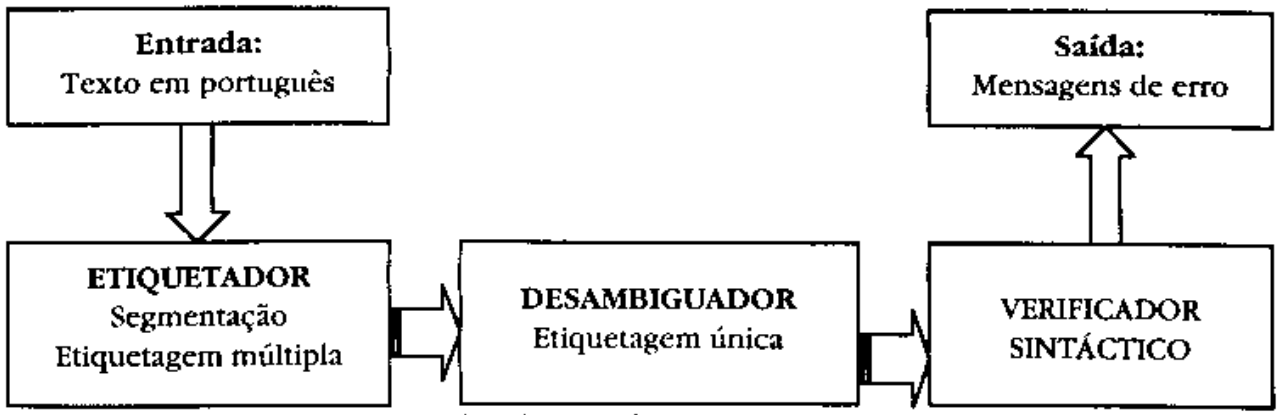


Figura 1. Arquitectura do Verificador Sintáctico

A opção metodológica subjacente à concretização do módulo de desambiguação consistiu na formalização de regras linguísticas baseadas nas propriedades distribucionais das palavras, de forma a garantir um maior rigor dos resultados a alcançar.

Tendo este projecto como finalidade o Processamento da Língua Natural (PLN) e visando a máxima correcção nos domínios da análise morfológica e sintáctica, com base em textos em suporte informático, teve-se em consideração as necessidades aplicacionais e/ou a gestão do tempo e dos recursos, para além de se considerar a importância de ter em conta o «conhecimento sobre o mundo», num trabalho para PLN, de forma a contornar certos aspectos do comportamento de uma língua natural, nomeadamente as irregularidades morfológicas e sintácticas, dificilmente colmatadas por outras abordagens.

O problema da ambiguidade categorial é relevante no domínio do PLN pois a sua resolução implica, ao nível da verificação sintáctica, a alocação de vastos recursos computacionais. Uma das funções do módulo de desambiguação consiste em «decidir», tendo como base a informação fornecida pelo Etiquetador morfossintáctico, que faz corresponder à palavra as etiquetas relativas às suas diversas categorias, qual a etiqueta a atribuir a determinada palavra perante um contexto sintáctico específico. Por exemplo, a sequência sublinhada da frase «*Esta baixa o vidro, enquanto aquela abre a porta.*» pode ser tratada, parcialmente, pelo seguinte conjunto de regras:

Regra 1: F → SN SV
Regra 2: SN → Det N
Regra 3: SN → Cli
Regra 4: SN → Dem Adj N
Regra 5: SN → Dem N
Regra 6: SN → Dem
Regra 7: SV → V SN

Tendo em conta a ambiguidade categorial, numa aplicação computacional destas regras, o sistema aplicaria a Regra 1, uma vez que a frase contém um SN e um SV. Seguidamente, passa a verificar as regras de SN. A Regra 2 para a verificação de [SN,F], o SN sujeito, não seria aplicada, uma vez que prevê a existência de um Det e um N, o que não se verifica no referido constituinte. As regras 3, 4 e 5 seriam verificadas sem sucesso. Aplicaria então a Regra 6 (o SN sujeito é constituído apenas por um DEM). Finalmente poderiam ser aplicadas as Regras 7 e 2, para a validação do [SV,F] e do [SN,SV], respectivamente, o que, e consequentemente, validaria a aplicação da Regra 1.

2. O módulo de desambiguação e pré-sintaxe

Para além de resolver a ambiguidade das palavras, o Desambiguador desempenha uma outra tarefa que pode ser caracterizada como processamento pré-sintáctico do texto. Com efeito, são construídos, durante a aplicação das regras de desambiguação, domínios gramaticais[i] que vão dividir o texto (lista de unidades linguísticas etiquetadas) em sub-listas particulares dentro das quais as unidades linguísticas estabelecem relações privilegiadas entre si (por exemplo, entre as unidades que constroem certos domínios, deverá existir concordância em género e número). Os domínios gramaticais (que não correspondem necessariamente aos constituintes sintagmáticos) vão constituir um primeiro tratamento sintáctico do texto, que poderá ou não ser reaproveitado numa gramática mais elaborada, e serão a base a partir da qual o processo de correcção vai operar.

Considera-se vantajosa a existência de um módulo de desambiguação e pré-sintaxe numa fase anterior à análise gramatical, uma vez que o módulo de correcção, que exige mais capacidade de processamento, receberá os dados já pré-processados, diminuindo em muito o número de análises sintácticas a realizar, o que aumenta, por consequência, o desempenho do sistema. Por outro lado, a natureza dos textos-alvo, que eventualmente contêm erros, implica a opção por uma metodologia de desambiguação robusta.

3. Enquadramento teórico e opções metodológicas

As regras linguísticas, baseadas nas propriedades distribucionais das palavras, depois de formalizadas, definem no seu conjunto a gramática de desambiguação.

A proposta que estrutura este projecto, que se inscreve nas correntes das gramáticas de superfície, valoriza a componente linguística visando também ultrapassar o estudo circunscrito a alguns fenómenos da língua e processar textos reais através de um formalismo que funcione como um meio e não como uma finalidade em si.

Apesar de partirem de uma abordagem linguística, as propostas das gramáticas de unificação[ii] não apresentam resultados satisfatórios[iii] aquando do processamento de textos reais.

Por outro lado, as abordagens estatísticas, que têm apresentado alguns resultados positivos, partindo de um investimento mínimo, revelam-se muito limitadas, uma vez que se baseiam somente em *corpora*, não recorrendo portanto ao conhecimento sobre o mundo. Não é no entanto garantido que os *corpora* contemplem todos os aspectos da língua, fazendo com que a «estatística» acabe por não dar conta de determinados comportamentos, não significando que estes não existam na língua natural.

Se se tiver como referência os trabalhos de CHANOD e TAPANAINEN (1995) e SAMUELSSON e VOUTILAINEN (1997) conclui-se que, embora exija um investimento superior em tempo, a introdução de conhecimento linguístico para a desambiguação de etiquetagem de texto apresenta resultados muito mais fiáveis.

4. Opções adoptadas na implementação do desambiguador

Numa fase inicial da elaboração das regras, e tendo em vista a verificação e validação das mesmas, foi desenvolvido um compilador da gramática de desambiguação em *Prolog*. Este protótipo de desambiguação categorial, que foi inicialmente implementado para o sistema operativo *Windows 95*, e posteriormente portado para *Linux*, aceita como informação de entrada a informação de saída transformada do Analisador Morfossintáctico — um mapeamento não injectivo entre o conjunto dos itens lexicais constitutivos do texto em tratamento e o conjunto das etiquetas —, devolvendo como resultado o conjunto de análises válidas.

Tendo a implementação informática do Desambiguador partido da ferramenta pré-existente — o Analisador —, a linguagem utilizada foi também a linguagem *C*, apesar de, e dada a especificidade deste módulo, ter sido necessário uma adaptação da infra-estrutura já existente. Para a construção do Desambiguador foi necessário ter em conta, previamente, uma fase de refinamento das etiquetas das categorias atribuídas pelo Analisador, uma vez que o comportamento distribucional de diferentes itens de uma mesma categoria não permitia, com as regras linguísticas, a desambiguação dessas etiquetas.

5. A gramática de desambiguação

A finalidade de uma gramática de desambiguação é atribuir a cada item lexical uma e só uma categoria. O formalismo desenvolvido para as regras de desambiguação exibe um formato semelhante ao da linguagem *Prolog*, linguagem esta usada no protótipo inicial do Desambiguador.

A implementação do Desambiguador em linguagem *C* permite obter, dada a informação de saída do Analisador, todos os resultados possíveis permitidos pelas regras de desambiguação desenvolvidas. É de salientar que o formalismo das regras é independente da implementação informática, garantindo ao linguista uma maior flexibilidade no seu desenvolvimento e a garantia de um funcionamento homogéneo das regras de desambiguação.

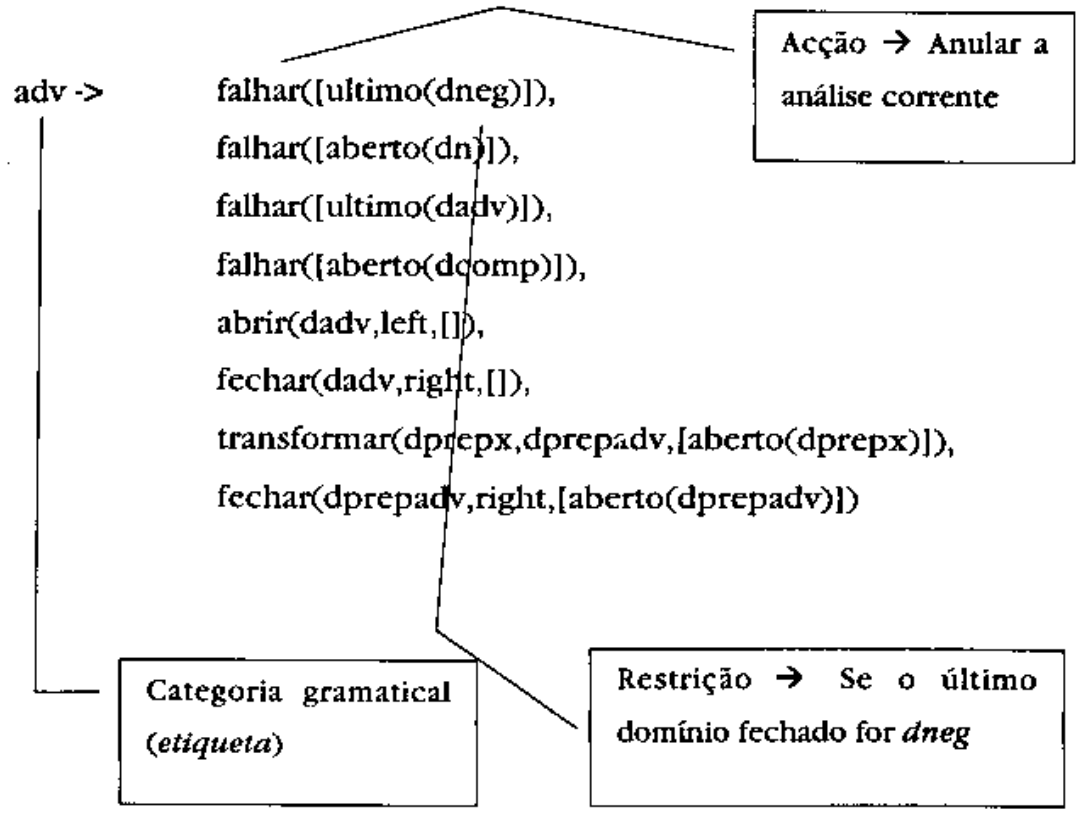


Figura 2. Exemplo de uma regra

As regras de desambiguação possuem uma forma semelhante ao de cláusulas *Prolog*, isto é, uma regra possui um conjunto de acções que são desencadeadas com o aparecimento das categorias se se verificarem as condições expressas na regra.

5.1 Descrição da gramática de desambiguação

regra	→ etiqueta -> condição.
etiqueta	→ *n* | *adj* | *adv* | *prep* | *neg* |... *(etiquetas do Analisador morfossintáctico)*
condição	→ acção | condição,acção
acção	→ *falhar* | *abrir* | *fechar* | *transformar* | *anulafecho*
falhar	→ *falhar*([restrições])
abrir	→ *abrir*(domínio,local,[restrições])
fechar	→ *fechar*(domínio,local,[restrições])
transformar	→ *transformar*(domínio,domínio,posição,[restrições])
anulafecho	→ *anulafecho*(domínio,posição,[restrições])

domínio	→ *dn* | *dv* | *dprep* | *dx* | *dprepx* | ...
local	→ *left* | *rigth*
posição	→ *laste* | *plast* | *alast*
restrições	→ [ ] | [restrição]
restrição	→ restrição_atómica | restrição,restrição_atómica
restrição_atómica	→*nexiste* |*naberto* |*aberto* |*ultimo* | *penultimo* |*nao*
nexiste	→ *nexiste*(posição)
ultimo	→ *ultimo*(domínio)
penultimo	→ *penultimo*(domínio)
aberto	→ *aberto*(domínio)
naberto	→ *naberto*(domínio)
nao	→ *nao*(posição,domínio)

5.2 Descrição algorítmica do mecanismo de aplicação das regras

Considere-se um parágrafo *P* com uma estrutura de lista cujos elementos são listas de itens. Regra geral, *P* terá apenas um elemento, todavia no caso de existir ambiguidade de segmentação poderá ter mais.

Nesta descrição exemplifica-se o processo de tratamento de apenas um elemento denominado *LI*. Seja *LI* uma lista de itens pertencente a *P* e *CA* o conjunto de análises. Tome-se o primeiro elemento de *LI* e seja CA_0 = [ ] — o conjunto de análises do início da execução —, localize-se a regra correspondente à etiqueta do item lexical por pesquisa no ficheiro de regras linguísticas; aplique-se cumulativamente as operações prescritas pela regra que deverão estar implementadas como funções booleanas — se alguma das operações devolver *false* a análise é abandonada. No final deste passo, deve ter-se obtido o conjunto CA_1 — conjunto de análises após tratamento do primeiro item lexical, ou seja, conjunto de análises no Momento 1. O processo é repetido para o segundo elemento de *LI*, sendo aplicado a cada uma das análises de CA_1 obtido no passo anterior. Assim, assiste-se a uma potencial explosão combinatorial do número de análises. Esta explosão é potencial, pois os predicados principais têm a dupla função de construção e filtragem das análises, pelo que algumas análises nunca chegam a ser geradas. Os predicados secundários, por seu turno, têm apenas a função de filtragem e são, tal como os principais, implementados como funções booleanas, tendo estes últimos como domínio de aplicação a última frase de cada análise. Os restantes elementos de *LI* são tratados de modo análogo ao segundo.

5.3. Exemplo do tratamento de uma frase

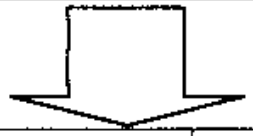
Tome-se como exemplo a seguinte frase: «*Na reunião serão debatidas as estratégias*». Após o tratamento efectuado pelo Analisador, é fornecida a seguinte informação, que vai servir de «entrada» para o Desambiguador.

[em]	prep		
[a]	art		
[reunião]	n		
[serão]	n	vaux	vcop2
[debatidas]	ppass	adj	
[as]	art	cli_ac	
[estratégias]	n		
[.]	pfinal		

Como se pode observar, três unidades lexicais (*serão, debatidas* e *as*) apresentam ambiguidade, isto é, apresentam mais do que uma etiqueta a elas associadas. Tendo em conta esta ambiguidade, seria possível existirem 12 estruturas diferentes, como se pode observar no Quadro 1, sendo apenas uma delas, neste caso[iv], correcta. O Desambiguador vai ter de as reduzir, apresentando apenas uma delas — a correcta —, e simultaneamente fazer um esboço de análise gramatical, agrupando as unidades lexicais em «unidades gramaticais», o que vai facilitar, posteriormente, a verificação sintáctica.

Quadro 1
Estruturas possíveis tendo em conta as etiquetas atribuídas pelo Analisador

em	*a*	*reunião*	*serão*	*debatidas*	*as*	*estratégias*	.
prep	art	n	n	ppass	art	n	pfinal
prep	art	n	n	ppass	cli_ac	n	pfinal
prep	art	n	n	adj	art	n	pfinal
prep	art	n	n	adj	cli_ac	n	pfinal
prep	***art***	***n***	***vaux***	***ppass***	***art***	***n***	***pfinal***
prep	art	n	vaux	ppass	cli_ac	n	pfinal
prep	art	n	vaux	adj	art	n	pfinal
prep	art	n	vaux	adj	cli_ac	n	pfinal
prep	art	n	vcop2	ppass	art	n	pfinal
prep	art	n	vcop2	ppass	cli_ac	n	pfinal
prep	art	n	vcop2	adj	art	n	pfinal
prep	art	n	vcop2	adj	cli_ac	n	pfinal



prep	art	n	vaux	ppass	art	n	pfinal

Observe-se, então, como o Desambiguador vai aplicando as regras, eliminando todas as estruturas erradas e apresentando apenas a que tem as etiquetas correctas, de acordo com o contexto em que cada uma das unidades lexicais ocorre.

A etiquetagem apresentada pelo Analisador, como foi acima referido, vai funcionar como informação de entrada para o Desambiguador, como se exemplifica a seguir.

(i)[**prep**(em)],[**art**(a)],[**n**(reunião)],[**n**(serão),**vaux**(serão),**vcop2**(serão)], [**ppass**(debatidas),**adj**(debatidas)],[**art**(as),**cli_ac**(as)],[n(estratégias)],[**pfinal**(.)]

As regras vão sendo aplicadas a cada unidade lexical, tendo em conta não só a etiqueta mas também as estruturas gramaticais que vai construindo.

A primeira unidade lexical da frase é *em*, à qual foi atribuída a etiqueta *prep*. Visto que esta unidade lexical está no início da frase, ainda não há qualquer estrutura a ser «construída». Então, o primeiro e único passo do Desambiguador é verificar todas as acções previstas na regra de *prep*, ou seja, não vai ter necesidade de considerar a(s) possível(eis) estrutura(s) já «construída(s)». Se aplicar a acção desconstrutora[v], a análise é anulada e já não é verificada qualquer outra acção. No caso específico da *prep* não é aplicado esse tipo de acção, visto que uma preposição pode iniciar uma frase[vi]. Não sendo, então, aplicada a acção desconstrutora, vão ser verificadas todas as acções construtoras e aplicadas todas as que validam as condições definidas nas diversas acções da regra de *prep*.

Sendo assim, à unidade lexical 1 — *em* —, à qual corresponde a etiqueta *prep*, vai ser aplicada a seguinte acção

(ii) ***abrir(dprepx,left,[naberto(dprep)]***

que se deve ler «abrir um domínio preposicional indefinido (*dprepx*) à esquerda (*left*), se não estiver aberto (*naberto*) nenhum domínio preposicional (*dprep*).

A partir deste momento, o Desambiguador começa a «construir» uma estrutura, que neste caso é a seguinte:

(iii) A. [**dprepx <prep(em)>**

Esta estrutura vai ser mantida para a aplicação da regra da unidade seguinte.

Ao encontrar a unidade lexical 2 — *a* — com a etiqueta *art*, e tendo em conta a estrutura que foi mantida após a aplicação da regra de *prep*, vai ser aplicada a regra de *art*. A acção que neste caso satisfaz as condições definidas é

(iv) ***transformar(dprepx,dprepn,[aberto(dprepx)])***

que diz que deve «transformar um domínio preposicional indefinido em domínio preposicional nominal (*dprepn*), se aquele estiver aberto». Se se observar a estrutura A, pode verificar-se que existe um *dprepx* aberto[vii], logo a restrição é satisfeita e esta acção da condição da regra é aplicada, concatenando a unidade

lexical à estrutura existente. Mais uma vez, visto que não foi aplicada a acção desconstrutora, a nova estrutura (que se apresenta a seguir) é mantida para que a análise continue.

(v) A. **[dprepn <prep(em)> <art(a)>**

A unidade lexical 3 é *reunião*. Esta unidade não apresenta qualquer ambiguidade, visto que lhe foi atribuída apenas a etiqueta *n* (cf. (i), acima). O Desambiguador aplica da regra de *n*, a acção

(vi)***fechar(dprepn,right,[aberto(dprepn)])***

Esta acção diz que deve «fechar um domínio preposicional nominal à direita (*right*), se estiver aberto». Observando a estrutura em (v), pode verificar-se que o mesmo se encontra aberto. A unidade lexical é concatenada e o domínio é fechado, como se exemplifica a seguir

(vii) A. **[dprepn <prep(em)> <art(a)> <n(reunião)> dprepn]**

Atente-se agora no que é que a unidade lexical 4 — *serão* — «desencadeia». Esta unidade lexical é ambígua (cf. (i), acima), pois apresenta três etiquetas diferentes — *n*, *vaux* e *vcop2*. Sendo ambígua, vai fazer com que cada uma das regras — de cada etiqueta — verifique todas as acções definidas na sua condição, tendo em conta a estrutura que manteve, apresentada em (vii).

A primeira etiqueta é *n*. Neste caso, a acção que vai ser aplicada é a seguinte

(viii)***falhar([ultimo(dprepn),naberto(dprepx)])***

Sendo um acção desconstrutora — «falhar a análise se o último domínio fechado for um domínio preposicional nominal e não estiver aberto um domínio preposicional indefinido» —, a possível estrutura a criar vai ser anulada, permitindo que não seja atribuída a etiqueta *n*, de acordo com o contexto, à unidade lexical *serão*.

Restam, agora, duas etiquetas. De seguida são verificadas as acções da regra de *vaux*. É aplicada a seguinte acção:

(ix) ***abrir(daux,left,[naberto(daux)])***

É uma acção construtora — «abrir um domínio auxiliar (*daux*) à esquerda, se não estiver aberto» —, então o domínio vai ser aberto, a unidade lexical vai ser concatenada e esta estrutura vai ser mantida.

(x) **[dprepn <prep(em)> <art(a)> <n(reunião)> dprepn] [daux <vaux(serão)>**

Finalmente, a etiqueta *vcop2*. Das acções definidas na condição da regra de *vcop2*, vão ser aplicadas as seguintes:

(xi)[viii] ***abrir(dvcop2,left,[naberto(dv)])***
fechar(dvcop2,right,[aberto(dvcop2)])

A par da estrutura de (x), vai ser mantida uma nova estrutura, uma vez que não foi aplicada, em relação à etiqueta *vcop2*, nenhuma acção desconstrutora.

(xii) B. **[dprepn <prep(em)> <art(a)> <n(reunião)> dprepn] [dvcop2 <vcop2(serão)> dvcop2]**

Ambas as estruturas vão ser mantidas para «prosseguir» a análise.

A unidade lexical *debatidas* apresenta duas etiquetas, *ppass* e *adj*. Vai ser necessário, então, a verificação da condição da regra de *ppass* e a condição da regra de *adj*, em relação a cada uma das estruturas mantidas (cf. (vii) e (ix), respectivamente). Tendo em conta a estrutura A (apresentada em vii), e a etiqueta *ppass*, as acções da condição da regra que vão ser aplicadas são as seguintes:

(xiii) ***abrir(dpart,left,[naberto(dpart)])***^{dx}
fechar(dpart,right,[aberto(dpart)])^{x}
fechar(daux,right,[aberto(daux)])^{xf}

Com a aplicação destas acções, a estrutura A passa a ser a seguinte:

(xiv) **A. [dprepn <prep(em)> <art(a)> <n(reunião)> dprepn] [daux <vaux(serão)> [dpart <ppass(debatidas)> dpart] daux]**

Não estava nenhum domínio participial (*dpart*) aberto, logo a restrição da primeira acção (da condição da regra de *ppss*) apresentada em (xiii) está de acordo com a estrutura que foi tida em conta (a A). Assim, foi aberto o domínio previsto pela acção. A segunda acção em (xiii), visto que é aplicada depois da anteriormente referida, vai encontrar um *dpart* aberto, permitindo assim que seja fechado. De seguida, e visto que a restrição da terceira acção é validada, vai ser fechado o domínio auxiliar (aberto com uma das acções da condição da regra de *vaux* (cf. (ix)).

Como foi referido acima, a unidade lexical 5 — *debatidas* —, é ambígua. Vai ser necessário verificar a condição da regra da outra etiqueta atribuída — *adj* —, ainda em relação à estrutura A. A acção definida na condição da regra de *adj* que vai ser aplicada é a que se apresenta a seguir.

(xv)***falhar([aberto(daux)])***

Neste «passo» vai ser aplicada a acção desconstrutora. A análise vai «falhar», uma vez que a restrição é «satisfeita»: há um domínio auxiliar aberto (cf. (x)). Mantém-se apenas a estrutura em (xiv).

As etiquetas da unidade lexical em causa foram até agora «verificadas» em relação à estrutura A. Mas é necesário que sejam também verificadas em relação à estrutura B (apresentada em (xii) e «construída» a partir da inicial (a A) com a aplicação da regra de *dvcop2*). Sendo assim, e tendo em conta então a estrutura B, da regra de *ppass* vai ser aplicada a acção (neste caso a desconstrutora),

(xvi)***falhar([ultimo(dvcop2)])***

que está de acordo com a restrição estabelecida: o último domínio fechado é um *dvcop2* (cf. (xii)), o que faz com que a etiqueta *ppass* seja «rejeitada» na frase em causa, não mantendo também uma possível estrutura com essa etiqueta. Ainda em relação à mesma unidade lexical e à mesma estrutura, mas tendo em conta a etiqueta *adj*, vai(ão) ser aplicada(s) a(s) acção(ões) cujas restrições estejam de

acordo com o contexto. Sendo assim, vão ser aplicadas as acções apresentadas em (xvii).

(xvii) ***abrir(dadj,left,[naberto(dn),naberto(dprepn)])***
fechar(dadj,right,[aberto(dadj)])

Se se observar as restrições, a acção deve permitir a abertura de um domínio adjectival à esquerda se não houver (*naberto*) nenhum domínio nominal ou preposicional nominal aberto. De seguida o domínio é fechado, visto que é o que está previsto pela segunda acção apresentada em (xvii) e uma vez que a restrição é «satisfeita» (se houver um domínio adjectival aberto - foi aberto pela acção anterior, a primeira de (xvii)). A estrutura B vai ser mantida como se exemplifica a seguir.

(xviii) B. **[dprepn <prep(em)> <art(a)> <n(reunião)> dprepn] [dvcop2 <vcop2(serão)> dvcop2] [dadj <adj(debatidas)> dadj]**

A unidade lexical 6 apresenta, tal como as duas anteriores, ambiguidade. Para cada uma das etiquetas, vai ser necessário verificar as respectivas regras em relação a cada estrutura já construída (até agora, A e B). Sendo assim, é aplicada, da regra de *n*, a acção

(xix) ***abrir(dn,left,[naberto(dprepx)])***

que vai abrir um domínio nominal (*d n*), visto que não há um domínio preposicional indefinido aberto. Com a aplicação desta regra, a estrutura A passa a ser

(xx) A. **[dprepn <prep(em)> <art(a)> <n(reunião)> dprepn] [daux <vaux(serão)> [dpart <ppass(debatidas)> dpart] daux] [dn <art(a)>**

À regra da etiqueta *cli_ac* corresponde a seguinte acção desconstrutora

(xxi) ***falhar([ultimo(daux)])***

a acção que vai ser aplicada tendo em conta a estrutura A (em (xiv)). Uma possível nova estrutura não é criada, sendo mantida apenas a que se apresenta em (xx).

A partir de B (apresentada em (xviii)), vão ser criadas duas novas estruturas, que se designam B.1 e B.2 e se apresentam a seguir:

(xxii) **B.1 [dprepn <prep(em)> <art(a)> <n(reunião)> dprepn] [dvcop2 <vcop2(serão)> dvcop2] [dadj <adj(debatidas)> dadj] [dn <art(as)>**

(xxiii) B.2 **[dprepn <prep(em)> <art(a)> <n(reunião)> dprepn] [dvcop2 <vcop2(serão)> dvcop2] [dadj <adj(debatidas)> dadj] [dcli_ac <cli_ac(as)> dcli_ac]**

B.1 é «criada» a partir da aplicação da acção da regra de *art* que verificam as restrições,

(xxiv) ***abrir(dn,left,[naberto(dprepx)])***

e B.2 de *cli_ac*,

(xxv) ***abrir(dcli_ac,left[naberto(dcli_ac)])***
fechar(dcli_ac,right,[abert(dcli_ac])

Neste momento da análise existem 3 estruturas que vão ser mantidas para a verificação da unidade lexical seguinte — *estratégias*.

A unidade lexical *estratégias* não apresenta ambiguidade, isto é, foi-lhe atribuída apenas uma etiqueta. Sendo assim, só a regra dessa etiqueta (*n*) vai ser verificada. À estrutura A. (em (xx)), é aplicada a acção

(xxvi)***fechar(dn,right,[aberto(dn)])***

que vai fechar o domínio nominal à direita (aberto com a acção em (xix)). Como se pode observar, a restrição da acção em (xxvi) é satisfeita — há um *dn* aberto (cf. (xx)).

Com a aplicação da acção em (xxvi) a estrutura A. passa a ser a que a seguir se apresenta.

(xxvii) **A. [dprepn <prep(em)> <art(a)> <n(reunião)> dprepn] [daux <vaux(serão)> [dpart <ppass(debatidas)> dpart] daux] [dn <art(a)> <n(estratégias)> dn]**

À estrutura B.1 é aplicada a mesma acção — (xxvi) —, uma vez que a restrição é, também, satisfeita, e passa a ser a seguinte:

(xxviii) B.1 **[dprepn <prep(em)> <art(a)> <n(reunião)> dprepn] [dvcop2 <vcop2(serão)> dvcop2] [dadj <adj(debatidas)> dadj] [dn <art(as)> <n(estratégias)> dn]**

A regra da etiqueta da unidade lexical 7 — *estratégias* — vai ser verificada ainda em relação à estrutura B.2. A acção aplicada é a desconstrutora, apresentada em (xxix), o que vai fazer anular a análise, mantendo, assim, para a verificação do último item, o ponto final (*pfinal*), apenas duas, a A. e a B.1 (cf. (xxvii) e (xxviii), respectivamente).

(xxix)***falhar([ultimo(dcli_ac),nao(plast,dv),naberto(dn),naberto(dpr epn)])***

A unidade, neste caso, não lexical, que termina a frase em análise é o ponto final. A esta unidade está também associada uma etiqueta e consequentemente uma regra. O Desambiguador, ao encontrar a etiqueta *pfinal*, vai verificar as acções previstas na condição da regra e aplicar a(s) que satisfaça(m) as restrições definidas.

Em relação à estrutura A., aplica as seguintes acções:

(xxx) ***abrir(dpont,left,[naberto(dpont)])***
fechar(dpont,right,[aberto(dpont)])

Com a sua aplicação, termina a análise e apresenta a estrutura em (xxxi).

(xxxi) A. **[dprepn <prep(em)> <art(a)> <n(reunião)> dprepn] [daux <vaux(serão)> [dpart <ppass(debatidas)> dpart] daux] [dn <art(a)> <n(estratégias)> dn] [dpont <pfinal(.)> dpont]**

Já em relação à estrutura B.1, é aplicada uma acção desconstrutora, o que faz com que a análise seja anulada. A acção aplicada é a seguinte:

(xxxii)***falhar([ultimo(dn),penultimo(dadj),antepenultimo(dvcop2)])***

ENTRADA:
[**prep**(em)], [**art**(a)], [n(reunião)], [n(serão),**vaux**(serão),**vcop2**(serão)], [**ppass**(debatidas),**adj**(debatidas)], [**art**(as),**cli_ac**(as)], [n(estratégias)], [**pfinal**(.)]

SAÍDA:
[**prep**(em)], [**art**(a)], [n(reunião)], [**vaux**(serão)], [**ppass**(debatidas)], [**art**(as)], [n(estratégias)], [**pfinal**(.)]

Considerações finais

Neste momento a desambiguação está a ser implementada recorrendo ao menor número de informação possível, ou seja, considerando apenas a informação contextual de cada etiqueta. Por exemplo, não existe, neste momento, diferenciação de género nem número, o que não permite desambiguar casos em que a verificação de concordância é essencial (Ex.: A cavalo dado não se olha o dente.).

São tratados casos de ambiguidade nome-adjectivo, nome-verbo e adjectivo--verbo. O tratamento dos verbos também é diferenciado consoante sejam copulativos, auxiliares ou principais, pois diferem não só nas suas distribuições categoriais como nos mecanismos de controlo de erros de concordância que irão ser desenvolvidos.

Tendo em conta as características do Desambiguador, certos aspectos da língua não poderão ser totalmente abrangidos (como é o caso da predicação secundária). Frases como «A rapariga viu o filme bonita» não serão tratadas de forma correcta apesar de ser possível assinalar estes casos com uma mensagem de aviso.

O Desambiguador tem como objectivo a possibilidade de obter textos etiquetados com um mínimo de ambiguidade categorial e, ao mesmo tempo, obter domínios gramaticais que irão tornar a tarefa do módulo seguinte - o Verificador Sintáctico - mais fácil e eficiente do ponto de vista linguístico e também do ponto de vista computacional, pois os textos tratados já possuem alguma informação sobre a estrutura sintáctica.

Notas

i Designa-se por *domínio gramatical* uma sequência de elementos que estabelecem entre si uma relação estreita, por exemplo, a flexão.

ii Sobre gramáticas de unificação, cf., por exemplo, POLLARD e SAG (1994)

iii Cf. HAGÈGE e BÈS (1998).

iv Nalguns casos pode haver mais do que uma interpretação, o que implica também mais do que uma análise possível.

v Cf. *3.1. Descrição da Gramática de Desambiguação*

vi No caso do português europeu, como exemplo de aplicação de uma acção desconstrutora em início de frase (entenda-se aqui como início de frase o início de um parágrafo ou o elemento lexical que segue um ponto), há a acção que não permite que ocorra uma unidade lexical com a etiqueta *cli* (clítico pronominal na sua forma acusativa ou dativa).

vii Neste texto, nas estruturas das análises efectuadas, «[» indica a abertura de um domínio e «]» indica o fecho.

viii *dv - domínio verbal*
dvcop2 - domínio verbal copulativo 2

ix «Abrir um domínio participial à esquerda se não estiver aberto.»

x «Fechar um domínio participial à direita se estiver aberto.»

xi «Fechar um domínio auxiliar à direita se estiver aberto.»

Bibliografia

AÏT-MOKHTAR, Salah (1995), *SMORPH: Guide d'utilisacion*. Rapport tecnhique, GRIL, Université Blaise Pascal, Clermont-Ferrand.

——————— (1997), «Du texte ASCII au texte lemmatisé: la présyntaxe en une seule étape», in *Actas de TALN'97 (Traitement Automatique du Langage Naturel)*, Grenoble.

ANDRADE, Ernesto d' (1993), *Dicionário Inverso do Português*, Lisboa: Edições Cosmos.

CHANOD, J.-P. e Pasi TAPANAINEN (1995), «Tagging French - comparing a statistical and a constraint-based method», in *Proceedings of the EACL-95*, Dublin.

COSTA, J. Almeida e A. Sampaio e MELO (1990), *Dicionário da Língua Portuguesa*, 6.ª edição revista e aumentada, Porto: Porto Editora.

CUESTA, Pilar Vázquez e Maria Albertina Mendes da LUZ (1971), *Gramática da Língua Portuguesa*, Lisboa: Edições 70.

CUNHA, Celso e Lindley CINTRA (1985), *Nova Gramática do Português Contemporâneo*, Rio de Janeiro: Editora Nova Fronteira.

FERREIRA, Aurélio Buarque de Holanda (1986), *Novo Dicionário da Língua Portuguesa*, 2.ª edição revista e aumentada, 2.ª impressão, Rio de Janeiro: Editora Nova Fronteira.

GREFENSTETTE, Gregory e Pasi TAPANAINEN (1994), «What is a word. What is a sentence? Problems of Tokenization», in *The Proceedings of the 3rd International Conference on Computational Lexicography (COMPLEX'94)*, Budapeste.

HAGÈGE, Caroline, António MEIRELES, Brígida TRINDADE, Carla DIOGO e Fernando LEITE (1997), *Analisador Morfossintáctico*, Relatório técnico, Instituto de Linguística Teórica e Computacional, Lisboa.

HAGÈGE, Caroline e Gabriel BÈS (1998), «Da observação de propriedades linguísticas à sua formalização numa gramática do processamento da língua», *Actas do III Encontro*

para o processamento computacional da língua portuguesa escrita e falada. Porto Alegre

MATEUS, Maria Helena Mira, Ana Maria BRITO, Inês DUARTE e Isabel Hub FARIA (1989), *Gramática da Língua Portuguesa*, 2.ª edição revista e aumentada, Série Linguística, Lisboa: Editorial Caminho.

POLLARD, Carl e Ivan A. SAG (1994), *Head-Driven Phrase Structure Grammar*, Chicago: CSLI.

SAMUELSSON, Christer e Atro VOUTILAINEN (1998), «Comparing a Linguistic and a Stochastic Tagger», *Actas do III Encontro para o processamento computacional da língua portuguesa escrita e falada*. Porto Alegre

SÁ NOGUEIRA, Rodrigo de (1991), *Dicionário de verbos portugueses conjugados*, 9.ª edição, Lisboa: Clássica Editora.

SANCHEZ LEON, Fernando (1995), «Development of a Spanish Version of the Xerox Tagger», Facultad de Filosofía y Letras, Universidade Autónoma de Madrid, Madrid.

SILVA, Emídeo e António TAVARES (1989), *Dicionário dos Verbos Portugueses: conjugação e referências*, Porto: Porto Editora.

TEYSSIER, Paul (1984), *Manuel de Langue Portugaise - Portugal-Brésil*, 12ème édition revue et corrigée, Paris: Editions Klincksieck.

VILLALVA, Alina (1994), *Estruturas Morfológicas: Unidades e Hierarquias nas Palavras do Português*, Dissertação de Doutoramento apresentada à FLUL.

VILELA, Mário (1990), *Dicionário do Português Básico*, Porto: Edições Asa.

WILKENS, Mike e Julian KUPIEC (1995), «Training Hidden Markov Models for Part of Speech Tagging». Xerox Corporation, Palo Alto.