Resolução de anáfora pronominal em português utilizando o algoritmo de Hobbs

Este exemplar corresponde à redação final da Dissertação devidamente corrigida e defendida por Denis Neves de Arruda Santos e aprovada pela Banca Examinadora.

Campinas, 19 de agosto de 2008.

Profa. Dra. Ariadne Maria Brito Rizzoni Carvalho
Instituto de Computação, Unicamp
(Orientadora)

Dissertação apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA DO IMECC DA UNICAMP
Bibliotecária: Maria Júlia Milani Rodrigues – CRB8a / 2116

Santos, Denis Neves de Arruda
Sa59r Resolução de anáfora pronomial em português utilizando o algoritmo de Hobbs / Denis Neves de Arruda Santos -- Campinas, [S.P. :s.n.], 2008.

Orientadora : Ariadne Carvalho
Dissertação (mestrado) - Universidade Estadual de Campinas, Instituto de Computação.

1. Algoritmos de computador. 2. Inteligência artificial. 3. Processamento de linguagem natural (Computação). I. Carvalho, Ariadne. II. Universidade Estadual de Campinas. Instituto de Computação. III. Título.

Título em inglês: Hobbs´ algorithm for pronoun resolution in portuguese.

Palavras-chave em inglês (Keywords): 1. Computer algorithms 2. Artificial intelligence. 3. Natural language processing (Computer science)

Área de concentração: Processamento de Língua Natural

Titulação: Mestre em Ciência da Computação

Banca examinadora:

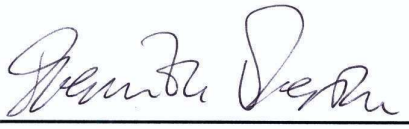
Profa. Dra. Ariadne Carvalho (IC-UNICAMP)
Profa. Dra. Renata Vigora (PUC-RS)
Prof. Dr. Tomasz Kowaltowski (IC-UNICAMP)

Data da defesa: 20/06/2008

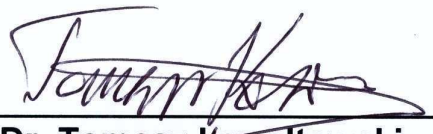
Programa de Pós-Graduação: Mestrado em Ciência da Computação

TERMO DE APROVAÇÃO

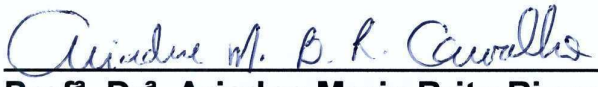
Dissertação Defendida e Aprovada em 20/06/2008, pela Banca examinadora composta pelos Professores Doutores:



Profª. Drª. Renata Vieira
Faculdade de Informática – PUC-RS



Prof. Dr. Tomasz Kowaltowski
IC – UNICAMP



Profª. Drª. Ariadne Maria Brito Rizzoni Carvalho
IC – UNICAMP

Instituto de Computação
Universidade Estadual de Campinas

Resolução de anáfora pronominal em português utilizando o algoritmo de Hobbs

Denis Neves de Arruda Santos[1]

Agosto de 2008

Banca Examinadora:

- Profa. Dra. Ariadne Maria Brito Rizzoni Carvalho
  Instituto de Computação, Unicamp (Orientadora)
- Profa. Dra. Renata Vieira
  Faculdade de Informática, PUC-RS
- Prof. Dr. Tomasz Kowaltowski
  Instituto de Computação, Unicamp
- Prof. Dr. Jacques Wainer
  Instituto de Computação, Unicamp (Suplente)
- Profa. Dra. Lucia Helena Machado Rino
  Centro de Ciências Exatas e de Tecnologia, UFSCar (Suplente)

[1]Suporte financeiro de: Bolsa do CNPq (processo 135017/2006–8) 2006–2008.

Resumo

Anáfora é uma referência abreviada a uma entidade, esperando que o receptor do discurso possa compreender a referência. A automatização da resolução de anáforas pode melhorar o desempenho de vários sistemas de processamento de língua natural, como tradutores, geradores e sumarizadores. A dificuldade no processo de resolução acontece nos casos em que existe mais de um referente possível. Pesquisas sobre a resolução de anáforas na língua portuguesa ainda são escassas, quando comparadas com as pesquisas para outras línguas, como por exemplo, o inglês. Este trabalho descreve uma adaptação para o português do algoritmo sintático proposto por Hobbs para resolução de anáfora pronominal. A avaliação foi feita comparando os resultados com os obtidos por outro algoritmo sintático para resolução de pronomes, o algoritmo de Lappin e Leass. Os mesmos corpora foram utilizados e uma melhora significativa foi obtida com o algoritmo de Hobbs.

Abstract

Anaphora is an abreviated reference to an entity expecting the receiver of the discourse can understand the reference. Automatic pronoun resolution may improve the performance of natural language systems, such as translators, generators and summarizers. Difficulties may arise when there is more than one potential candidate for a referent. There has been little research on pronoun resolution for Portuguese, if compared to other languages, such as English. This paper describes a variant of Hobbs' syntactic algorithm for pronoun resolution in Portuguese. The system was evaluated comparing the results with the ones obtained with another syntactic algorithm for pronoun resolution handling, the Lappin and Leass' algorithm. The same Portuguese corpora were used and significant improvement was verified with Hobbs' algorithm.

Agradecimentos

Considerando esta dissertação como resultado de uma caminhada que não começou na Unicamp, agradecer pode não ser uma tarefa fácil. Para evitar injustiças, agradeço de antemão a todos que de alguma forma passaram pela minha vida e contribuíram para a formação de quem sou hoje.

E agradeço, particularmente, a algumas pessoas pela contribuição direta na construção deste trabalho:

Aos meus pais, meu irmão e a toda minha família, que, com muito carinho, não mediram esforços para que eu chegasse até esta etapa de minha vida.

Aos meus tios Paulo, Cristina, Silas e Célida, pelo apoio moral e estímulo que me permitiram levar este trabalho até ao fim.

À professora Dra. Ariadne Carvalho, pela amizade, pela paciência na orientação e pelo incentivo que tornaram a conclusão desta dissertação possível.

Ao professor Dr. Tomasz Kowaltowski e à professora Dra. Renata Vieira pelas valiosas contribuições.

Aos demais professores doutores da casa, pelos conhecimentos transmitidos.

Ao prof. Arnaldo Mandel, pela preciosa colaboração.

Ao amigo Thiago Coelho, pela paciência e ajuda.

Ao amigo Fábio Bezerra, com quem tive o prazer de trocar experiências e saberes díspares.

Agradeço ao CNPq, pelo apoio financeiro.

Cabe, aqui, também um agradecimento aos amigos Renato, Humberto, Paulo, Clewton e Paula pelo companheirismo e incentivo.

Sumário

Lista de Tabelas

Lista de Figuras

Capítulo 1

Introdução

O processamento automático da língua natural é um ramo da inteligência artificial que tem por objetivo interpretar e gerar textos em língua natural [3]. Trata-se de uma tarefa extremamente difícil para um computador realizar [28]. A principal dificuldade é a ambigüidade que existe nas línguas naturais. Enquanto um ser humano consegue facilmente obter o significado de uma sentença[1], dentre um conjunto de possíveis interpretações, é menos provável que um computador consiga, devido a inúmeros fatores, tais como a falta de habilidade para lidar com contextos complexos.

A ambigüidade pode ocorrer no nível léxico, com o uso de palavras que possuem mais de um significado, como "arquivo" e "banco" [28]. Também pode ocorrer no nível sintático, quando é possível criar mais de uma estrutura sintática para a mesma sentença, como em "A menina viu o menino com o telescópio". Além disso, ela pode aparecer no nível semântico, como em "João viajará até abril".

A coesão é um fenômeno da língua que, por meio de mecanismos (referência, substituição, elipse, conjunção e coesão lexical), determina uma relação semântica entre um elemento do texto e algum outro elemento crucial para a sua interpretação [19, 23].

A referência é um elemento que não pode ser interpretado isoladamente, porque remete a outro item do discurso necessário à sua interpretação. Esse outro item é chamado de referente. A sentença (1), a seguir, exemplifica o uso de referências.

(1) As posições de **Bento XVI**$_i$ vão afastar católicos, mas **ele**$_i$ possui argumentos corajosos e interessantes.[2]

O elemento "ele" é uma referência ao elemento "Bento XVI", o referente, ou antecedente [20].

[1] Neste trabalho, o termo sentença é utilizado como sinônimo de frase e oração.

[2] Índices iguais indicam correferência.

O processo de resolução consiste na tarefa de determinar o referente. A dificuldade está nos casos em que existe mais de um candidato a referente [13], como no exemplo da sentença (2), a seguir

(2) João culpou Pedro por **ele** ter batido seu carro.

onde o pronome "ele" pode se referir tanto a "João" como a "Pedro". Essa sentença é ambígua até para os seres humanos.

Algumas sentenças podem parecer ambíguas sintaticamente, mas um leitor pode ter uma interpretação preferencial, que determina a escolha do referente. Na sentença (2), por exemplo, o verbo "culpar" pode fazer com que "Pedro" seja preferencialmente escolhido como antecedente.

Na sentença (3), pelo mesmo motivo, "Walter" pode ser escolhido como referente:

(3) Walter apresentou João para a mulher que atualmente é **sua** esposa.

Existem várias formas de referência. A Figura 1.1 apresenta os tipos de referência encontrados na língua:

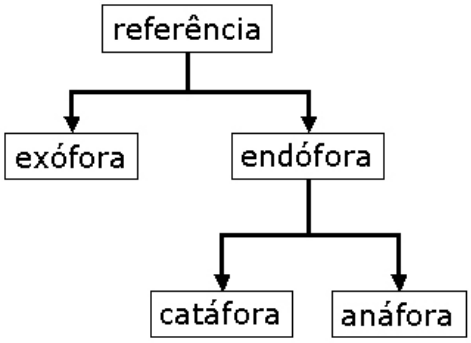


Figura 1.1: Classificação de referência.

Existem dois tipos de referência: a exófora e a endófora. A exófora é aquela que remete a algum elemento fora do texto, como, por exemplo, o pronome "você" na sentença (4)

(4) Quando **você** acordar, diga "bom dia" a todos.

onde a informação requerida para interpretação do pronome "você" não é encontrada no texto, mas sim na situação comunicativa.

Na endófora, que é o objeto de estudo desta dissertação, o referente está expresso no texto. Ela pode ser de dois tipos: catáfora e anáfora. Na catáfora o referente aparece após a referência, como, por exemplo, o pronome "Ele" na sentença (5)

(5) **Ele**$_i$ era tão bom, **o meu marido**$_i$![3]

onde o referente "o meu marido" aparece após o pronome "Ele".

Em textos, quando uma referência é encontrada, é natural que ela se refira a algum termo citado anteriormente. Nos casos em que ocorre a catáfora, fica claro para o leitor que o termo se refere a um elemento que será apresentado depois [19].

Na anáfora, o referente aparece antes da referência. Uma anáfora acarreta a volta ao texto anterior para compreender a referência [25]. A sentença (6), a seguir, exemplifica o seu uso.

(6) O escritório de **Alice**$_i$ fica perto de **uma padaria**$_j$. **Ela**$_i$ costuma lanchar todo dia **lá**$_j$.

O termo "ela" é uma anáfora que referencia "Alice", que foi citado anteriormente. A forma adverbial "lá" é uma anáfora que referencia "uma padaria".

A endófora também pode ser classificada como intra- e inter-sentencial. Na referência intra-sentencial, o referente aparece na mesma sentença que a referência, como no caso da sentença (3). Na referência inter-sentencial, o referente não aparece na mesma sentença que a referência, como no caso da sentença (6).

A referência pronominal é uma das mais comuns e utiliza pronomes para criar a referência [29]. O pronome "ela", na sentença (6), é um exemplo de referência pronominal.

Um ser humano facilmente entende que o pronome na sentença (6) se refere a "Alice"; porém, é difícil para um computador determinar se o referente é "Alice" ou "uma padaria", dado que ambos têm o mesmo gênero e número (feminino e singular).

Existem casos em que o referente é uma sentença inteira, como, por exemplo, na sentença (7)

(7) Você disse que não gosta de animais, mas eu duvido **disso**.

onde o pronome "isso" se refere à sentença "você não gosta de animais".

Existem casos de referência em que o conhecimento pragmático precisa ser levado em consideração, como, por exemplo na sentença (8)

(8) João cortou a corda que suspendia a **caixa**$_i$ e **ela**$_i$ caiu.[4]

onde o pronome "ela" tem como antecedente o termo "caixa". Essa escolha depende da compreensão da conseqüência de "cortar a corda que suspendia a caixa".

[3]Exemplo extraído de Koch [23].
[4]Exemplo extraído de Chaves [9].

A automatização do processo de resolução de referências é importante para o desempenho de vários sistemas de processamento de língua natural, como tradutores, geradores, sumarizadores, processamento de diálogos e recuperação de informação [25].

Soluções encontradas para automatizar a resolução de referências podem contribuir para outras áreas de estudo da língua, tais como resolução da ambigüidade e coesão textual.

Vários algoritmos para resolução automática de anáforas já foram propostos, como o de Hobbs [21], Centering [5] e Lappin e Leass [24].

Hobbs usou duas técnicas para a resolução de anáforas pronominais: a semântica e a sintática. A semântica supõe que alguns conhecimentos gerais estão disponíveis na forma de predicados lógicos; inferências são realizadas com os predicados para interpretar o contexto das palavras, determinar a relação entre as sentenças e resolver a anáfora [21]. A técnica sintática é percorrer a árvore de derivação do texto procurando por uma entidade com gênero e número correspondentes à referência [28].

O objetivo desta dissertação foi avaliar o desempenho do algoritmo sintático de Hobbs na resolução de referências pronominais, em português.

Para tanto, o algoritmo foi adaptado para o português e avaliado na resolução de pronomes pessoais (retos e oblíquos), em quatro corpora diferentes. Os resultados da avaliação foram comparados com os obtidos por outro algoritmo sintático para resolução de pronomes, o algoritmo de Lappin e Leass [13], também adaptado para o português. Os mesmos corpora foram utilizados e uma melhora significativa foi obtida com o algoritmo de Hobbs.

Sabe-se que informações sintáticas não fornecem uma solução completa para o problema da resolução de anáfora [21, 6], mas a proposta era avaliar o algoritmo utilizando apenas informações sintáticas e descobrir até que ponto informações sintáticas contribuem para a resolução de referências pronominais.

O restante deste trabalho está organizado da seguinte forma:

No capítulo 2 são apresentados os trabalhos relacionados;

No capítulo 3 são apresentadas as ferramentas utilizadas no projeto;

No capítulo 4 são descritos os corpora utilizados para avaliação do algoritmo;

No capítulo 5 é descrito o algoritmo sintático de Hobbs para resolução de anáforas, assim como sua adaptação para o português;

No capítulo 6 são apresentados os resultados da avaliação do algoritmo, que por sua vez são comparados com os obtidos com o algoritmo de Lappin e Leass;

Finalmente, no capítulo 7 são apresentadas as conclusões e sugestões para trabalhos futuros.

Capítulo 2

Trabalhos relacionados

Neste capítulo, apresentamos alguns trabalhos relacionados à resolução de anáforas pronominais. Durante o desenvolvimento da pesquisa, percebemos a escassez de trabalhos para a língua portuguesa, se comparados aos trabalhos com excelentes resultados para o inglês.

O algoritmo de Lappin e Leass para resolução de anáforas pronominais em terceira pessoa está entre os principais algoritmos criados na década de 90 [24]. Ele foi desenvolvido para lidar com textos em inglês e utiliza informações sintáticas geradas por um analisador sintático[1].

O algoritmo realiza, para a escolha do antecedente de uma anáfora, o cálculo de pesos, que são atribuídos a cada sintagma nominal[2], de acordo com a estrutura sintática da sentença e de acordo com uma representação simples do modelo do discurso.

Os autores do algoritmo realizaram testes utilizando textos relativamente simples - manuais de computador - e obtiveram uma taxa de acerto de 86%. Os autores também realizaram o mesmo teste com o algoritmo de Hobbs e a taxa de sucesso foi 4% menor.

Uma adaptação do algoritmo de Lappin e Leass para o português foi tema da dissertação de mestrado de Coelho [11, 13, 12]. Coelho avaliou o algoritmo utilizando três corpora distintos: jurídico, literário e jornalístico. O corpus jurídico é composto por 16 Pareceres da Procuradoria Geral da República de Portugal, e é formado por sentenças longas e complexas. Para esse corpus o algoritmo teve sucesso em 35,15% das anáforas. O corpus literário é composto da obra "O Alienista", de Machado de Assis [2], e é formado por sentenças um pouco mais simples. Com esse corpus, a taxa de acerto foi de 32,61%. O terceiro corpus é composto por 14 textos jornalísticos, e possui as sentenças mais simples. Com esse corpus o algoritmo obteve 43,56% de acerto.

[1] Em inglês: *parser*.

[2] Sintagmas são grupos de elementos lingüísticos classificados de acordo com a categoria sintática do seu núcleo [3]. Os sintagmas nominais têm um substantivo como núcleo; os sintagmas verbais têm um verbo ou uma locução verbal como núcleo, etc.

É importante ressaltar que, apesar da taxa de acerto da adaptação do algoritmo de Lappin e Leass para o português ter sido bastante inferior à taxa de acerto do algoritmo original, a avaliação do algoritmo em inglês foi realizada com textos de natureza bem mais simples - manuais de computador.

Kennedy e Boguraev [22] propuseram uma adaptação do algoritmo de Lappin e Leass sem a necessidade de uma análise sintática completa do texto. O algoritmo proposto necessita apenas de algumas informações sintáticas e morfológicas dos sintagmas nominais presentes no texto. O analisador sintático utilizado possui um conjunto de padrões que é utilizado para determinar a função sintática dos sintagmas nominais. Os autores também adaptaram algumas regras utilizadas no cálculo dos pesos atribuídos aos sintagmas nominais. O algoritmo obteve uma taxa de acerto de 75%. A diferença no desempenho, dentre outros fatores, é devida ao uso de textos mais complexos que os utilizados por Lappin e Leass: foram utilizados 27 textos variados, como artigos de revistas, notícias e conteúdo da internet.

Palomar et al. [31] desenvolveram um algoritmo para resolução de pronomes pessoais de terceira pessoa, pronomes demonstrativos, pronomes reflexivos e pronomes ocultos em textos em espanhol. O algoritmo utiliza uma lista de restrições e preferências e também faz uso de uma lista de restrições de correferência para pronomes em espanhol. Nos testes realizados com textos literários, a taxa de sucesso foi de 76,8%. Para avaliar o desempenho do algoritmo proposto, os autores também implementaram o algoritmo de Hobbs, Lappin e Leass, Centering e um quarto algoritmo, chamado Proximity, que foi proposto pelos autores e é baseado em restrições e preferência pelo antecedente mais próximo. O principal algoritmo proposto no artigo teve um desempenho melhor que os quatro outros algoritmos implementados. As taxas de sucesso dos outros algoritmos foram: 62,7% para o algoritmo de Hobbs, 67,4% para o algoritmo de Lappin e Leass, 52,9% para o algoritmo Proximity e 62,6% para o algoritmo de Centering.

Parabone e Lima [32] propuseram um algoritmo para resolução de pronomes possessivos de terceira pessoa em português. O algoritmo utiliza informação sintática, semântica e conhecimento pragmático. Para avaliação do algoritmo, foram utilizadas as leis brasileiras de proteção ao meio-ambiente, tendo sido obtida uma taxa de acerto de 92,97%. A alta taxa de acerto se deve, dentre outros fatores, ao fato do algoritmo lidar apenas com anáforas intra-sentenciais, o que reduz a quantidade de candidatos a antecedentes, e ao fato do algoritmo ter sido desenvolvido para lidar com textos de um domínio específico.

Outro algoritmo bastante popular para resolução de anáforas é o algoritmo de Centering, proposto por Brennan et al. [5], que é derivado da Teoria de Centering, proposta por Grosz et al. [17]. Esse algoritmo foi implementado em um sistema chamado HPSG, que funciona como uma interface em língua natural para consultas a banco de dados.

A principal idéia da Teoria de Centering é que certas entidades mencionadas em uma

sentença são mais importantes que outras; esse fato impõe certas restrições ao uso de referências e, em especial, ao uso de pronomes [28].

A Teoria de Centering divide um discurso em segmentos, sendo que cada segmento possui uma entidade principal, chamada de centro[3]. A cada segmento do discurso está associado um centro retrospectivo (*backward-looking center*), que corresponde a entidade que está no centro do segmento, e um centro prospectivo (*forward-looking center*), que corresponde a um conjunto ordenado das entidades mencionadas e candidatas a centro no próximo segmento. As entidades do centro prospectivo são ordenadas de acordo com regras baseadas na função sintática das entidades: primeiro os sujeitos, depois os objetos e, por último, as demais entidades.

No algoritmo de Centering, inicialmente cria-se uma lista dos candidatos a antecedente formada pelas entidades mencionadas no discurso. Em seguida, os itens da lista são filtrados de acordo com restrições propostas na Teoria de Centering. Depois, a lista é ordenada com base em restrições da Teoria de Centering, de forma que os candidatos mais prováveis sejam os primeiros itens da lista.

Walker [33] avaliou o algoritmo de Centering e de Hobbs para o inglês utilizando três corpora. Dois deles foram também utilizados por Hobbs na avaliação de seu algoritmo [21]: o primeiro capítulo da novela "*Wheels*", de Arthur Haley, e a edição de 7 de julho de 1975 da revista "*Newsweek*". O terceiro corpus utilizado foi um diálogo entre duas pessoas sobre a montagem de uma bomba de água plástica. Com a novela, a taxa de acerto do algoritmo de Hobbs foi 88%, e a do algoritmo de Centering foi 93%. Com os artigos do jornal, o algoritmo de Hobbs acertou 89% e o de Centering acertou 84%. Já com o diálogo, o algoritmo de Centering teve um desempenho bem superior (64%) ao de Hobbs (51%).

Aires et al. [1] avaliaram o algoritmo de Centering para resolução de pronomes em português. O corpus utilizado foi o mesmo corpus jurídico utilizado por Coelho [11] na adaptação do algoritmo de Lappin e Leass e, de acordo com os autores, o algoritmo acertou 51% das resoluções.

Mitkov [28] propôs um algoritmo para resolver anáforas pronominais utilizando informações sintáticas e semânticas. Nesse algoritmo, inicialmente cria-se uma lista com todos os sintagmas nominais da sentença corrente e das duas sentenças anteriores que concordem em gênero e número com o pronome. Em seguida, os elementos da lista são pontuados de acordo com um conjunto de heurísticas. O sintagma nominal que possuir a maior pontuação ou, em caso de empate, o mais próximo da anáfora, será escolhido como referente. Na avaliação da sua proposta, Mitkov utilizou vários manuais técnicos que continham 223 anáforas pronominais no total. A taxa de sucesso obtida foi 89,7% (200 pronomes resolvidos corretamente).

[3]Em inglês: *center*.

Chaves [9] desenvolveu uma adaptação do algoritmo de Mitkov [28] para resolução de pronomes pessoais de terceira pessoa em textos em português. Nessa abordagem, a busca foi ampliada de duas para até três sentenças anteriores, e a lista de heurísticas foi modificada. A autora avaliou o algoritmo utilizando os mesmos corpora utilizados por Coelho [11] na adaptação do algoritmo de Lappin e Leass. As taxas de sucesso foram superiores ao de Lappin e Leass: 67,01% no corpus jornalístico, 38% no corpus literário e 54% no corpus jurídico.

Apesar do algoritmo de Mitkov ter apresentado um resultado superior ao algoritmo proposto por Coelho, esses resultados ainda são bem inferiores aos obtidos com o algoritmo original de Mitkov. Segundo a autora, isso se justifica, principalmente, pelos erros inseridos pelas ferramentas utilizadas no pré-processamento dos corpora.

No próximo capítulo, são descritas as ferramentas utilizadas na avaliação do algoritmo proposto.

Capítulo 3

Ferramentas utilizadas

Nesta seção, apresentamos as ferramentas utilizadas no trabalho. O produto final das ferramentas são arquivos XML que servem de entrada para o sistema desenvolvido.

3.1 PALAVRAS

Neste projeto utilizamos o analisador sintático PALAVRAS[1], desenvolvido no Departamento de Língua e Comunicação da *University of Southern Denmark*, em Odense, por Eckhard Bick e sua equipe [4].

Resultado de vários anos de pesquisa, o PALAVRAS possui ótimas estatísticas de acerto: mais de 99% para a análise morfológica e quase 97% para a análise sintática.

Este analisador é bastante robusto e chega a processar aproximadamente 400 palavras por segundo em um Pentium II de 300MHz com sistema Linux, e é ideal para ser utilizado por aplicações como marcação de corpus e tradutores automáticos.

Para exemplificar o uso desta ferramenta, considere a sentença (9):

(9) Pedro$_i$ bebeu cerveja$_j$. Ele$_i$ a$_j$ comprou no supermercado.

A Figura 3.1 mostra o resultado da análise realizada: um código que descreve a árvore sintática. Cada linha de código corresponde a um nó da árvore, sendo que o sinal de igual no início da linha indica o nível do nó na árvore.

[1] http://visl.sdu.dk/

```
UTT:cl(fcl)
S:prop('Pedro' M S)     Pedro
P:v('beber' fin  PS 3S IND)   bebeu
Od:n('cerveja' F S)     cerveja
.
UTT:cl(fcl)
S:pron('ele' pers  M 3S NOM)  Ele
Od:pron('ela' pers  F 3S ACC) a
P:v('comprar' fin  PS 3S IND) comprou
A:g(pp)
=H:prp('em' <sam->)     em
=D:g(np)
==D:art('o' <artd> <-sam> M S)      o
==H:n('supermercado' M S)     supermercado
```

Figura 3.1: Análise sintática gerada pelo PALAVRAS para a sentença (9).

O primeiro símbolo em cada linha representa a função sintática da palavra. Alguns destes símbolos são: 'S' para sujeito, 'P' para predicado , 'Od' para objeto direto, 'A' para adjunto adverbial, 'H' para núcleo do sintagma e 'D' para termos dependentes do sintagma.

Após o símbolo ':', é indicada a categoria morfossintática. Algumas das notações utilizadas são: 'cl' para oração[2], 'fcl' para oração finita[3], 'prop' para nome próprio, 'v' para verbo, 'pp' para sintagma preposcionado, 'np' para sintagma nominal, 'num' para número, 'n' para nome, 'adj' para adjetivo, 'art' para artigo, 'M' para masculino, 'F' para feminino, 'P' para plural, 'S' para singular, 'PS' para passado simples e 'IND' para indicativo. Por exemplo, "Pedro" é um nome próprio (prop), masculino (M) e singular (S); "comprou" é o verbo "comprar", conjugado na terceira pessoa do singular (3S), do passado simples (PS), do modo indicativo (IND).

3.2 Xtractor

Para facilitar a extração de informação do código gerado pelo PALAVRAS, Gaspering [16] desenvolveu uma ferramenta, chamada de Xtractor, que converte a saída do analisador PALAVRAS para XML.

A ferramenta gera três arquivos:

- Um arquivo *words*: que contém as palavras do texto;
- Um arquivo POS: que contém as informações morfológicas;

[2]Em inglês: *clause*.
[3]Em inglês: *finite clause*.

- Um arquivo *chunk*: que contém as informações sintáticas.

No arquivo *words*, exemplificado na Figura 3.2, cada palavra da sentença é associada a um identificador único ("id"). Esse identificador é utilizado nos demais arquivos.

```
<words>
 <word id="word_1">Pedro</word>
 <word id="word_2">bebeu</word>
 <word id="word_3">cerveja</word>
 <word id="word_4">.</word>
 <word id="word_5">Ele</word>
 <word id="word_6">a</word>
 <word id="word_7">comprou</word>
 <word id="word_8">em</word>
 <word id="word_9">o</word>
 <word id="word_10">supermercado</word>
 <word id="word_11">.</word>
</words>
```

Figura 3.2: Trecho do arquivo *words* para a sentença (9).

No arquivo POS, exemplificado na Figura 3.3, a classificação morfológica de cada palavra é indicada pela segunda *tag* dentro da *tag* "word". Por exemplo, a primeira *tag* se refere a palavra com identificador "word_1" e a segunda *tag* "prop" informa que se trata de um nome próprio masculino (gender="M") e singular (number="S").

```
<words>
 <word id="word_1">
  <prop canon="Pedro" gender="M" number="S"/>
 </word>
 <word id="word_2">
  <v canon="beber">
   <fin tense="PS" person="3S" mode="IND"/>
  </v>
 </word>
 <word id="word_3">
  <n canon="cerveja" gender="F" number="S" />
 </word>
  ...
</words>
```

Figura 3.3: Trecho do arquivo *POS* com as informações morfológicas.

No arquivo *chunk*, exemplificado na Figura 3.4, a função sintática de cada elemento ("chunk") é dada pelo atributo *ext*. O atributo *form* indica a classificação morfossintática, e o atributo *span* contém o identificador das palavras correspondentes. Por exemplo, o elemento "chunk_1" indica que existe uma sentença de "word_1" à "word_3"

(span="word_1..word_3"). A tag "ext" indica que a sentença é um enunciado declarativo[4]. O elemento "chunk_2" indica que o sujeito (ext="subj") da sentença é "word_1" (span="word_1") que, por sua vez, é um nome próprio (form="prop").

```
<text>
 <paragraph id="paragraph_1">
  <sentence id="sentence_1" span="word_1..word_4">
   <chunk id="chunk_1" ext="sta" form="fcl" span="word_1..word_3">
    <chunk id="chunk_2" ext="subj" form="prop" span="word_1"></chunk>
    <chunk id="chunk_3" ext="p" form="v_fin" span="word_2"></chunk>
    <chunk id="chunk_4" ext="acc" form="n" span="word_3"></chunk>
   </chunk>
  </sentence>
  ...
</paragraph>
</text>
```

Figura 3.4: Trecho do arquivo *chunk* com as informações sintáticas.

3.3 MMAX

O MMAX[5] (*Multi-Modal Annotation in XML*) [30] foi utilizado para marcação manual das anáforas e seus antecedentes. Essa ferramenta utiliza os arquivos *chunk* e *words*, produzidos pelo Xtractor, para gerar um arquivo XML de *markable* separado do corpus com as informações das marcações.

No arquivo de *markable*, exemplificado na Figura 3.5, o atributo "span" contém o identificador de uma palavra da sentença, e o atributo "form" indica se a marcação é de um pronome ou antecedente. Se for de um pronome, o atributo "pointer" indica a marcação do seu antecedente. Por exemplo, a marcação "markable_3" indica que a palavra correspondente à "word_5" é uma referência pronominal que referencia a marcação identificada como "markable_1". A marcação "markable_1", por sua vez, indica que "word_1" é um antecedente.

```
<markables>
 <markable id="markable_1" span="word_1" form="antecedent" />
 <markable id="markable_2" span="word_3" form="antecedent" />
 <markable id="markable_3" span="word_5" form="pronome"  pointer="markable_1" />
 <markable id="markable_4" span="word_6" form="pronome"  pointer="markable_2" />
</markables>
```

Figura 3.5: Trecho do arquivo *markable* com a marcação das referências e antecedentes.

[4]Em inglês: *statement*.

[5]http://mmax.eml-research.de

3.4 Problemas encontrados

Durante o desenvolvimento do trabalho, percebemos algumas erros nas informações morfológicas e sintáticas fornecidas pelo PALAVRAS. Todos os erros identificados foram corrigidos para que o algoritmo de Hobbs fosse melhor avaliado.

Um dos casos de erro morfológico ocorreu na sentença (10), que foi extraída do corpus jornalístico

(10) "Entre os presos estavam **três** petistas".

onde a palavra "três" foi identificada como substantivo masculino plural, ao invés de numeral.

Outro caso de erro morfológico apareceu na sentença (11) do corpus literário

(11) "Se a miséria viesse de chofre, o pasmo de Itaguaí seria enorme; mas **veio** devagar".

onde a palavra "veio" foi identificada como substantivo masculino singular, ao invés de verbo.

O analisador sintático PALAVRAS identificou os pronomes reflexivos "se" com informações de número, e, às vezes, de gênero. Na sentença (12), extraída do corpus jornalístico, por exemplo, o pronome reflexivo foi identificado como masculino:

(12) "Filhos, em geral, têm no pai o seu modelo e querem sempre equiparar-**se** a ele.".

Com o objetivo de realizar uma avaliação do algoritmo de Hobbs de forma que o resultado fosse independente do analisador sintático utilizado, as informações de gênero e número dos pronomes "se" foram desconsideradas.

Existem casos em que o PALAVRAS identificou como pronome palavras que, na verdade, eram índices de indeterminação do sujeito ou conjunções [7]. Na sentença (13), extraída do corpus literário, por exemplo, o índice de indeterminação do sujeito foi identificado como pronome reflexivo:

(13) "Trata-**se** de uma experiência científica".

Na sentença (14), extraída do corpus literário, o artigo destacado foi identificado como pronome pessoal.

(14) "**Os** sintomas de duplicidade e descaramento deste barbeiro são positivos".

Na mesma sentença, o substantivo "sintomas" foi identificado como feminino. Assim, após a correção, a palavra "os" foi removida da lista de referências pronominais marcadas.

A sentença (15) contém mais um exemplo de erro na identificação morfológica. O termo destacado foi identificado como advérbio:

(15) "Não há glamour nenhum **em menores** vivendo e se drogando em bueiros.".

Na sentença (16), o analisador PALAVRAS identificou "dele" como um pronome pessoal, apesar de ser um pronome possessivo. Ocorrências desse tipo foram eliminadas da lista de referências pronominais marcadas [15].

(16) Simão era médico. Um dos tios **dele** tinha uma fazenda.

Além dos problemas encontrados com as informações morfológicas, às vezes a estrutura da árvore sintática gerada pelo palavras também apresentava algum problema. A Figura 3.6[6] mostra um exemplo de árvore sintática com problema.

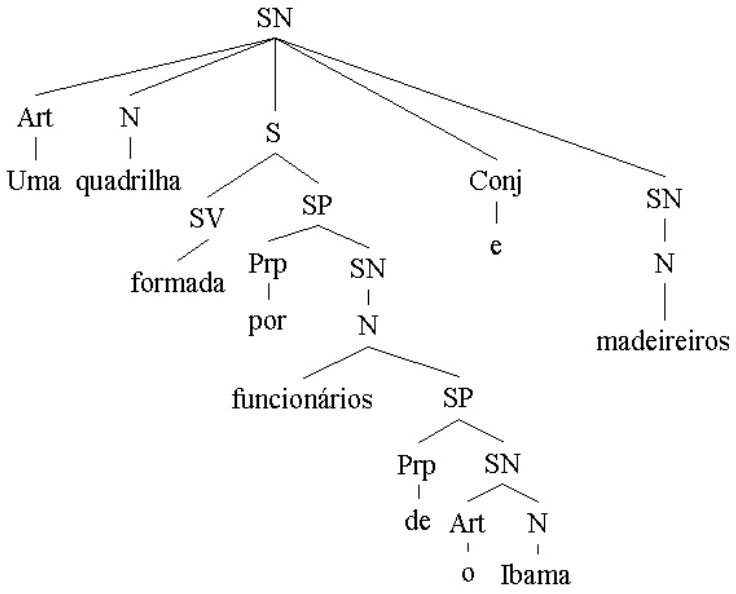


Figura 3.6: Exemplo de árvore sintática com problema.

A conjunção "e" e o sintagma nominal "madeireiros" não estão posicionados corretamente na árvore. A Figura 3.7 mostra a árvore sintática após a correção.

[6]O nó *S* corresponde à sentença, o nó *SN* corresponde ao sintagma nominal, o nó *SV* ao sintagma verbal, o nó *SP* ao sintagma preposicionado, o nó *Prp* à preposição, o nó *Art* ao artigo, o nó *Adj* ao adjetivo, o nó *N* ao substantivo, o nó *SADV* ao sintagma adverbial, o nó *Adv* ao advérbio, o nó *Pron* ao pronome relativo, o nó *Indf* ao pronome indefinido, o nó *Num* ao número e o nó *Conj* à conjunção.

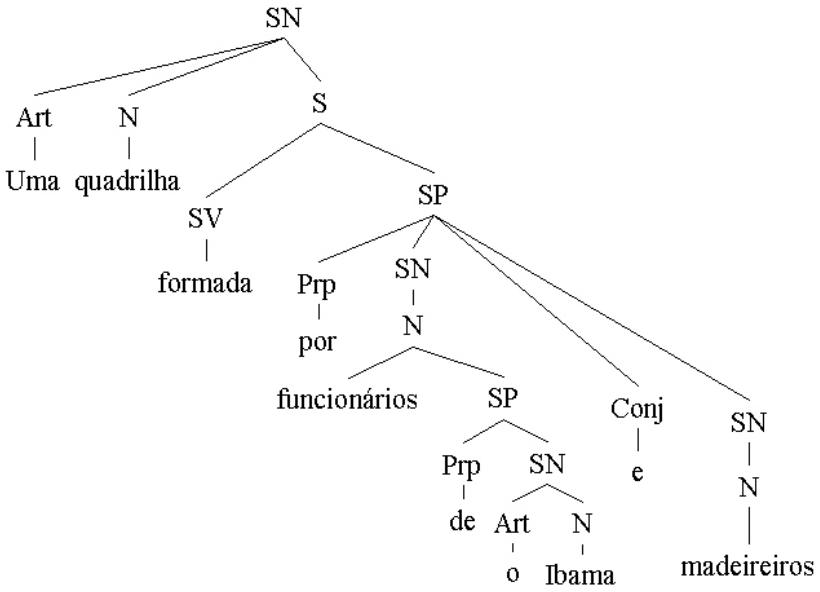


Figura 3.7: Exemplo de árvore sintática corrigida.

Outra dificuldade apresentada pelo PALAVRAS é que ele não realiza uma marcação para indicar quais sintagmas nominais compõem um sintagma composto. Visando a resolução de referências pronominais a sujeitos compostos, um processamento automático é utilizado para realizar essa marcação. O procedimento necessário já havia sido implementado por Coelho [11] e foi reutilizado neste trabalho. Esse procedimento consiste em agrupar os sintagmas nominais que compõem um sujeito composto.

Considere a sentença (17), a seguir:

(17) Maria e Pedro feriram-se.

A Figura 3.8 mostra a árvore gerada pelo PALAVRAS para a sentença (17) e a Figura 3.9 mostra o resultado depois da execução do pré-processamento.

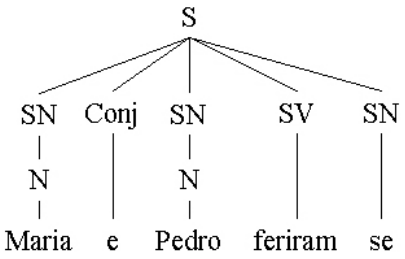


Figura 3.8: Árvore gerada pelo PALAVRAS para sujeito composto.

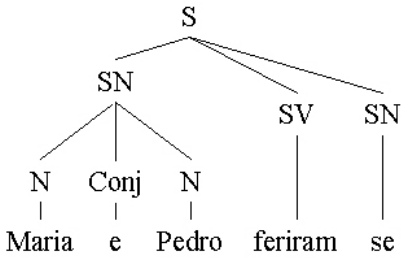


Figura 3.9: Árvore gerada depois do processamento do sujeito composto.

3.5 Considerações

Todas as correções das informações geradas pelo PALAVRAS foram realizadas de forma consistente, ou seja, foram realizadas mesmo que causassem a escolha do referente errado pelo algoritmo de Hobbs.

Como já foi afirmado neste capítulo, após algumas correções, palavras que haviam sido identificadas erroneamente como referências pronominais foram removidas da lista de marcações. Da mesma forma, pronomes que deixaram de ser marcados foram incluídos na lista.

Existe uma nova versão do analisador PALAVRAS[7] disponível; porém, após a realização de alguns testes, percebemos que vários dos problemas encontrados ainda permaneciam e, portanto, mantivemos a versão originalmente utilizada.

No próximo capítulo, são descritos os corpora utilizados na avaliação do algoritmo de Hobbs.

[7]http://beta.visl.sdu.dk/

Capítulo 4

Descrição dos corpora

Nesta dissertação, foram utilizados quatro corpora para avaliar o algoritmo. Três deles são os mesmos utilizados em Coelho [11]: um jornalístico, um literário e um corpus jurídico. O último corpus utilizado, chamado Summ-it [14], é composto por textos jornalísticos.

A seguir descreveremos cada um deles.

4.1 Corpus jornalístico

O corpus jornalístico é constituído por 14 textos, sendo um dos que possuem as sentenças mais simples.

Em Coelho [11], o corpus possuia 225 pronomes; porém, após algumas correções nos arquivos gerados pelo PALAVRAS e Xtractor, o número de pronomes foi reduzido para 171, sendo 81 (47,36%) reflexivos. A marcação desse corpus incluiu apenas os pronomes de terceira pessoa. A Tabela 4.1 mostra a freqüência de cada pronome.

Tabela 4.1: Freqüência dos pronomes no corpus jornalístico.

Pronome	Freqüência
se	80 (46,78%)
si	1 (0,58%)
o	4 (2,34%)
a	2 (1,17%)
lo	5 (2,92%)
los	1 (0,58%)
la	5 (2,92%)
las	2 (1,17%)
ele	30 (17,54%)
eles	13 (7,60%)
ela	14 (8,19%)
elas	11 (6,43%)
lhe	3 (1,75%)
Total	171

A Tabela 4.2 mostra a quantidade de referências intra- e inter-sentenciais.

Tabela 4.2: Quantidade de referências no corpus jornalístico.

Anáfora	Quantidade
Intra-sentencial	109 (63,74%)
Inter-sentencial	62 (36,26%)
Total	171

4.2 Corpus literário

O corpus literário consiste do livro "O Alienista", de Machado de Assis [2].
Esse corpus é de natureza complexa, como mostra o fragmento a seguir:

(18) "A Casa Verde foi o nome dado ao asilo, por alusão à cor das janelas, que pela primeira vez apareciam verdes em Itaguaí. Inaugurou-se com imensa pompa; de todas as vilas e povoações próximas, e até remotas, e da própria cidade do Rio de Janeiro, correu gente para assistir às cerimônias, que duraram sete dias."

Após o processamento automático, foram identificados 696 pronomes; porém, após realizarmos algumas correções manuais, o número de pronomes foi reduzido para 590, sendo 122 (20,67%) reflexivos. A marcação desse corpus incluiu apenas os pronomes de terceira pessoa. A Tabela 4.3 mostra a freqüência de cada pronome.

Tabela 4.3: Freqüência dos pronomes no corpus literário.

Pronome	Freqüência
se	109 (18,47%)
si	13 (2,20%)
o	65 (11,02%)
os	15 (2,54%)
a	35 (5,93%)
as	4 (0,68%)
lo	22 (3,73%)
los	4 (0,68%)
la	19 (3,22%)
las	1 (0,17%)
ele	112 (18,98%)
eles	10 (1,69%)
ela	22 (3,73%)
elas	3 (0,51%)
lhe	146 (24,75%)
lhes	10 (1,69%)
Total	590

A Tabela 4.4 mostra a quantidade de referências intra- e inter-sentenciais.

Tabela 4.4: Quantidade de referências no corpus literário.

Anáfora	Quantidade
Intra-sentencial	248 (42,03%)
Inter-sentencial	342 (57,97%)
Total	590

4.3 Corpus jurídico

O corpus jurídico é composto por 16 pareceres da Procuradoria Geral da República de Portugal. A anotação das anáforas pronominais não englobou todos os pronomes de terceira pessoa, e não incluiu os pronomes reflexivos.

O corpus jurídico é o mais complexo, composto de sentenças longas, como a sentença a seguir:

(19) "O casamento não irá afectar as legítimas expectativas dos filhos já existentes, visto que quer eles, quer os eventuais e futuros irmãos germanos, serão herdeiros de ambos os progenitores, quaisquer que sejam os bens, próprios ou comuns, destes, qualquer que seja o regime de bens convencionado."

Em Coelho [11], esse corpus possuia 297 pronomes; porém, após realizarmos algumas correções manuais, o número de pronomes foi reduzido para 287. A Tabela 4.5 mostra a freqüência de cada pronome.

Tabela 4.5: Freqüência dos pronomes no corpus jurídico.

Pronome	Freqüência
o	4 (1,39%)
os	1 (0,35%)
a	6 (2,09%)
as	3 (1,05%)
lo	17 (5,92%)
los	6 (2,09%)
la	9 (3,14%)
las	3 (1,05%)
ele	50 (17,42%)
eles	31 (10,80%)
ela	30 (10,45%)
elas	26 (9,06%)
no	1 (0,35%)
lhe	65 (22,65%)
lhes	35 (12,20%)
Total	287

A Tabela 4.6 mostra a quantidade de referências intra- e inter-sentenciais.

Tabela 4.6: Quantidade de referências no corpus jurídico.

Anáfora	Quantidade
Intra-sentencial	218 (75,96%)
Inter-sentencial	69 (24,04%)
Total	287

4.4 Corpus Summ-it

O corpus Summ-it foi elaborado para ser utilizado em pesquisas sobre discurso e sumarização automática. Esse corpus é constituido de 50 textos jornalísticos do caderno de Ciências da Folha de São Paulo, retirados do corpus PLN-BR (Recursos e Ferramentas para a Recuperação de Informação em Bases Textuais em Português do Brasil)[1].

O Summ-it foi criado devido à necessidade de um corpus anotado que possa servir como um padrão para a avaliação de sistemas automáticos de resolução de correferência para o português.

Esse corpus possui anotações de termos correferentes, que formam uma cadeia de correferência. A sentença (20) é um exemplo, extraído do corpus:

(20) **Pesquisadores do Museu Nacional do Rio de Janeiro**$_i$ anunciaram ontem a descoberta de uma nova espécie de dinossauro ... **os pesquisadores**$_i$ conseguiram estimar que o bicho fosse um filhote de 1,5 metro de altura.

O corpus também foi anotado com informações de relações retóricas, descritas na Teoria RST (*Rhetorical Structure Theory*) [27], o que resulta em uma estrutura arbórea chamada árvore RST.

O Summ-it possui 154 referências de pronomes pessoais marcadas; porém, como pretendemos avaliar a resolução de pronomes somente de terceira pessoa, o número de pronomes foi reduzido para 143. Essa marcação incluiu apenas 1 pronome reflexivo. A Tabela 4.7 mostra a freqüência de cada pronome.

[1] http://www.inf.unisinos.br/~renata/laboratorio/plnbr_index.htm

Tabela 4.7: Freqüência dos pronomes no corpus Summ-it.

Pronome	Freqüência
si	1 (0,70%)
o	6 (4,20%)
os	5 (3,50%)
a	4 (2,80%)
lo	5 (3,50%)
los	5 (3,50%)
la	2 (1,40%)
las	3 (2,10%)
ele	49 (34,27%)
eles	29 (20,28%)
ela	18 (12,59%)
elas	12 (8,39%)
na	1 (0,70%)
lhe	2 (1,40%)
lhes	1 (0,70%)
Total	143

A Tabela 4.8 mostra a quantidade de referências intra- e inter-sentenciais.

Tabela 4.8: Quantidade de referências no corpus Summ-it.

Anáfora	Quantidade
Intra-sentencial	59 (41,26%)
Inter-sentencial	84 (58,74%)
Total	143

4.5 Corpora total

Considerando os quatro corpora, temos um total de 1191 pronomes, sendo 204 (17,13%) reflexivos. A Tabela 4.9 mostra a freqüência de cada pronome.

Tabela 4.9: Freqüência dos pronomes nos corpora.

Pronome	Freqüência
se	189 (15,87%)
si	15 (1,26%)
o	79 (6,63%)
os	21 (1,76%)
a	47 (3,95%)
as	7 (0,59%)
lo	49 (4,11%)
los	16 (1,34%)
la	35 (2,94%)
las	9 (0,76%)
ele	241 (20,24%)
eles	83 (6,97%)
ela	84 (7,05%)
elas	52 (4,34%)
no	1 (0,08%)
na	1 (0,08%)
lhe	216 (18,14%)
lhes	46 (3,86%)
Total	1191

A Tabela 4.10 mostra a quantidade de referências intra- e inter-sentenciais.

Tabela 4.10: Quantidade de referências nos corpora.

Anáfora	Quantidade
Intra-sentencial	634 (53,23%)
Inter-sentencial	557 (46,77%)
Total	1191

4.6 Considerações

Neste capítulo apresentamos os quatro corpora utilizados na avaliação da adaptação do algoritmo de Hobbs. Três deles são os mesmo utilizados em Coelho [11]. Algumas correções precisaram ser realizadas nesses corpora devido aos erros gerados pelas ferramentas utilizadas, conforme descrito no capítulo 3.

No próximo capítulo, o algoritmo de Hobbs, assim como sua adaptação para o português, são descritos em detalhes.

Capítulo 5

Algoritmo de Hobbs

O algoritmo sintático de Hobbs [21] resolve referências pronominais intra- e inter-sentenciais. A resolução é realizada através de uma busca em largura, na árvore sintática da sentença, da esquerda para a direita, procurando por sintagmas nominais compatíveis em gênero e número com o pronome. Esse algoritmo lida com catáforas, mas não trata pronomes reflexivos, nem sentenças que são elas mesmas referentes [21].

A busca em largura é utilizada porque ela se mostrou mais eficiente para a maioria dos exemplos estudados por Hobbs. O fato da ordem de busca ser da esquerda para a direita favorece a escolha do sujeito da sentença, porque é mais provável que o pronome se refira ao sujeito do que ao objeto.

Esse algoritmo está entre os mais importantes criados entre as décadas de 60 a 80 [28], por ter apresentado um ótimo desempenho com a língua inglesa (mais de 90% de acerto), apesar de ser bastante simples. Ele é freqüentemente utilizado como *benchmark* para avaliar novas propostas [9].

Por questões de conveniência, Hobbs supõe que os pronomes são imediatamente dominados[1] por um nó *SN*.

O algoritmo recebe como entrada a árvore sintática da sentença e o nó correspondente à referência. Ele é composto dos seguintes passos:

1. Começar pelo nó *SN*, que imediatamente domina o pronome, e subir na árvore até encontrar um nó S ou *SN*. Chamar este nó de X, e o caminho percorrido para chegar até ele de p;

2. Realizar uma busca em largura, da esquerda para a direita, em todos os filhos de X que estejam à esquerda do caminho p. Retorne como antecedente o primeiro nó *SN* encontrado que seja compatível com o pronome, e que contenha um nó *SN* ou S entre ele e X;

[1] Dizemos que um nó A é dominado por todos os seus ancestrais.

3. Enquanto X não for o nó S mais alto da sentença:

   (a) A partir do nó X, subir na árvore até encontrar um nó S ou SN. Chamar este nó de X, e o caminho percorrido para chegar até ele de p;

   (b) Se X for um nó SN compatível com o pronome, e se o caminho p não passar pelo nó N que X imediatamente domina, retorne X;

   (c) Realizar uma busca em largura, da esquerda para a direita, em todos os filhos de X que estejam à esquerda do caminho p, e retorne o primeiro nó SN encontrado que seja compatível com o pronome;

   (d) Se X for um nó S, realizar uma busca em largura, da esquerda para a direita, em todas as arestas de X que estejam à direita do caminho p, mas não ir para baixo de nenhum nó SN ou S encontrado. Retorne o primeiro nó SN encontrado que seja compatível com o pronome;

4. Percorrer a árvore das sentenças anteriores do texto, começando pelas sentenças mais próximas. Em cada árvore é realizada uma busca em largura, da esquerda para a direita. Retorne o primeiro nó SN encontrado que seja compatível com o pronome.

Se a referência pronominal aparecer no mesmo nível na árvore que o referente, então provavelmente essa referência será um pronome reflexivo. Porém, como o algoritmo de Hobbs não resolve pronomes reflexivos, no passo 2 do algoritmo existe uma restrição que evita a escolha de um nó que esteja no mesmo nível que a referência na árvore. Esse restrição determina que um nó SN só poderá ser escolhido como referente se existir um nó SN ou S entre ele e a referência. Considere as seguintes sentenças:

(21) Pedro o cortou.

(22) Pedro se cortou.

De acordo com o passo 2, o pronome "o" na sentença (21) não pode se referir ao sujeito "Pedro" pois, se fosse o caso, um pronome reflexivo deveria ser utilizado, como na sentença (22).

Para exemplificar o funcionamento do algoritmo, considere a sentença (23), que possui uma anáfora intra-sentencial, e cuja árvore de derivação é mostrada na Figura 5.1:

(23) **Edinho**$_i$ escapou de um assalto depois que os ladrões **o**$_i$ reconheceram.

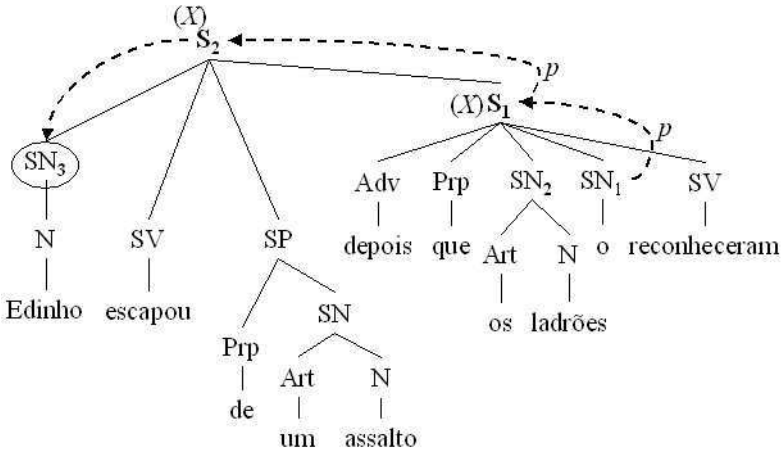


Figura 5.1: Resolução de anáfora intra-sentencial.

De acordo com o passo 1, a execução começa no nó SN_1, que imediatamente domina o pronome, e sobe para o nó S_1. O nó S_1 é chamado de X, e o caminho percorrido, de p. De acordo com o passo 2, é realizada uma busca em largura nos filhos de X à esquerda de p, e o nó SN_2 é encontrado. Porém, ainda de acordo com o passo 2, como o nó SN_2 não possui um outro nó S ou SN entre ele e X, ele é descartado. Como S_1 não é o nó mais alto da sentença, de acordo com o passo 3.a o nó S_2 é visitado. O passo 3.b não se aplica, porque o nó S_2 não é um SN. De acordo com o passo 3.c, é realizada uma busca em largura nos filhos de S_2, e o nó SN_3 é encontrado e aceito como antecedente do pronome "o".

Outra consideração importante sobre a árvore sintática é que um nó N, filho de um nó SN, pode possuir um sintagma preposicionado (SP) como filho, conforme proposto por Chomsky [10]. A sentença (24) exemplifica um caso em que essa condição é útil:

(24) **Costa**$_i$ achou um meio de provar que não **lhe**$_i$ cabia prisão.

A Figura 5.2 demonstra a execução do algoritmo para resolução do pronome "lhe" na sentença (24).

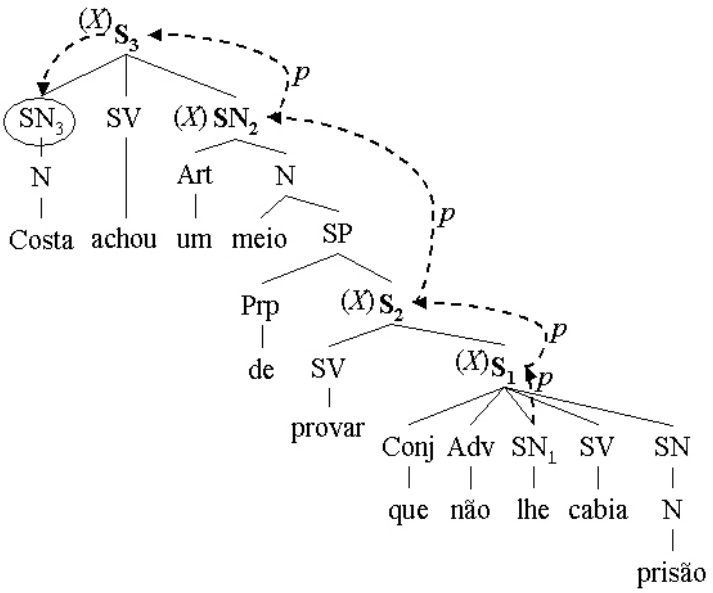


Figura 5.2: Exemplo de árvore sintática com nó SP abaixo de N.

De acordo com o passo 1, a execução começa pelo nó SN_1, sobe para S_1, e o caminho percorrido é chamado de p. De acordo com o passo 2, é realizada uma busca em largura nos filhos de S_1, à esquerda do caminho p, mas nenhum nó SN é encontrado. Como S_1 não é o nó mais alto da sentença, de acordo com o passo 3.a o nó S_2 é visitado. O passo 3.b não se aplica, porque S_2 não é um nó SN. Em seguida, de acordo com o passo 3.c, é realizada uma busca em largura nos filhos de S_2, mas não há nó SN à esquerda do caminho p. De acordo com o passo 3.d, é realizada uma busca nos filhos de S_2 à direita do caminho p, mas não existem arestas à direita do caminho p. Como S_2 não é o nó mais alto da sentença, a execução volta para o passo 3.a, e o nó SN_2 é visitado. De acordo com o passo 3.b, verifica-se que SN_2 é compatível com o pronome em número; porém, como o caminho p passa através do nó N que SN_2 domina imediatamente, SN_2 é rejeitado. Em seguida, de acordo com o passo 3.c, é realizada uma busca em largura nos filhos de SN_2, mas não há nó SN à esquerda do caminho p. De acordo com o passo 3.d, é realizada uma busca nos filhos de SN_2 à direita do caminho p, mas não existem arestas à direita do caminho p. Como SN_2 não é o nó mais alto da sentença, a execução volta para o passo 3.a, e o nó S_3 é visitado. O passo 3.b não se aplica, porque S_3 não é um nó SN. Em seguida, de acordo com o passo 3.c, é realizada uma busca em largura nos filhos de S_3, e o nó SN_3 é encontrado e aceito como referente.

Considere a sentença (25), a seguir, que contém uma anáfora inter-sentencial:

(25) **O parlamentar**$_i$ foi denunciado. **Ele**$_i$ será investigado.

A Figura 5.3 mostra a resolução de anáfora inter-sentencial contida na sentença (25).

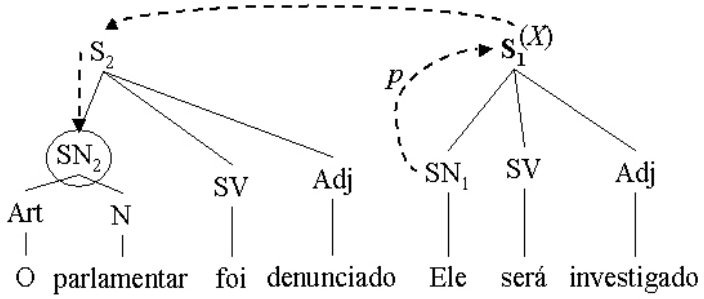


Figura 5.3: Resolução de anáfora inter-sentencial.

De acordo com o passo 1, a execução começa pelo nó SN_1, subindo para o nó S_1, e o caminho percorrido é chamado de p. De acordo com o passo 2, é realizada uma busca em largura nos filhos de S_1, à esquerda do caminho p, mas nenhum nó SN é encontrado. De acordo com o passo 3, como S_1 é o nó mais alto da sentença, a execução passa para o passo 4, e o nó S_2 é visitado. É realizada uma busca em largura, da esquerda para a direita, nos filhos de S_2, e o nó SN_2 é aceito como antencedente do pronome "Ele".

No exemplo (25), pode-se observar que, quando o referente do pronome não é encontrado na sentença atual, de acordo com o passo 4 é feita uma busca nas sentenças anteriores, começando pelas mais próximas. Hobbs [21] recomenda que o número de sentenças anteriores investigadas seja limitado, e cita o trabalho de Charniak [8] que sugere um janela de cinco sentenças. Entretanto, nos testes realizados por Hobbs, foram encontrados antecedentes em até nove sentenças anteriores.

Se, após atingir o limite do número de sentenças a ser investigadas, nenhum antecedente for encontrado, o algoritmo falha.

O passo 3.d do algoritmo lida com os casos de catáfora em que o referente comanda[2] o pronome. Além disso, nesse passo existe uma restrição para não procurar abaixo de nenhum nó SN ou S encontrado. Essa restrição foi incluída por Hobbs porque ela contribuiu para a melhoria do desempenho do algoritmo nos exemplos estudados por ele.

A Figura 5.4 exemplifica a resolução do pronome "o", que é uma catáfora:

(26) O cão seguia-$\mathbf{o}_i$ para todo o lado, reparou **o rapaz**$_i$ quando se voltou.[3]

[2]O nó SN_1 comanda SN_2 se um não domina o outro, e se o nó S que imediatamente domina SN_1 também domina SN_2 (mas não imediatamente) [21].

[3]Exemplo extraído de <http://www.fcsh.unl.pt/edtl/verbetes/C/catafora.htm>. Acessado em 15 de março de 2007.

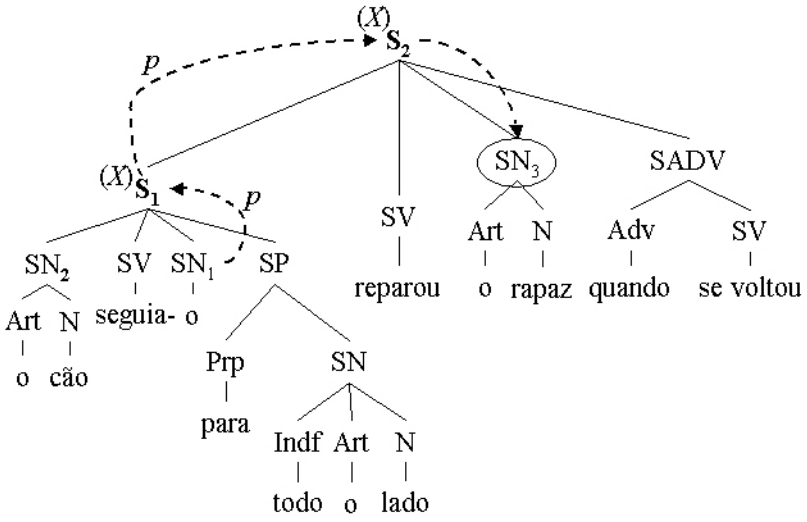


Figura 5.4: Resoluçao de catáfora.

De acordo com o passo 1, a execução começa pelo nó SN_1 e sobe para S_1, que passa a ser chamado de nó X, e o caminho utilizado para chegar até X é chamado de p. De acordo com o passo 2, é realizada uma busca em largura nos filhos de X à esquerda de p, e o nó SN_2 é encontrado. Porém, ainda de acordo com o passo 2, como o nó SN_2 não possui um outro nó S ou SN entre ele e X, ele é descartado. Como S_1 não é o nó S mais alto da sentença, de acordo com o passo 3.a o nó S_2 é visitado. Esse nó é chamado de X, e o caminho percorrido, de p. O passo 3.b não se aplica porque o nó X não é um SN. O passo 3.c não retorna nada, porque não existem nós à esquerda do caminho p. Finalmente, de acordo com o passo 3.d, após a busca em largura à direita do caminho p, SN_3 é encontrado e aceito como antecedente.

Em diálogos, Hobbs supõe que, antes do algoritmo ser executado, os emissores e ouvintes já foram descobertos. Assim, para a resolução de pronomes de terceira pessoa encontrados em trechos entre aspas, o emissor e o ouvinte do discurso devem ser descartados como possíveis antecedentes.

Hobbs propõe que se o mesmo pronome anafórico ocorrer duas vezes na mesma sentença, ou em duas sentenças consecutivas, as anáforas devem possuir o mesmo antecedente. Essa heurística foi denominada de "pronomes repetidos". Nos testes realizados por ele, essa heurística se mostrou pouco eficaz. Ela foi utilizada 48 vezes e retornou o antecedente correto apenas em 28 dos casos (58,3%).

Hobbs testou o algoritmo utilizando três corpora diferentes: o livro "*Early Civilization in China*" de William Watson (páginas de 21 à 69); o primeiro capítulo da novela "*Wheels*" de Arthur Haley (páginas de 1 à 6); e a edição de 7 de julho de 1975 da revista "*Newsweek*" (páginas de 13 à 19), começando pelo artigo "*A Ford in High Gear*". O algoritmo foi

aplicado em 100 ocorrências consecutivas de referências pronominais em cada corpus.

Segundo o autor, o algoritmo funcionou para 88,3% dos casos. Um dos motivos que levaram a essa taxa de sucesso, segundo ele, é um princípio seguido pelo ramo editorial de que todos os pronomes devem ter um antecedente. Nos textos utilizados, foi encontrado apenas um caso cuja sentença era ela mesma o antecedente.

Como um recurso adicional do algoritmo, existe um mecanismo capaz de manipular um conjunto de restrições semânticas que funcionam como um filtro. Dessa forma, qualquer entidade do texto que seja sugerida como antecedente deverá ser filtrada por esse mecanismo. Ainda de acordo com o autor, quando o algoritmo utilizou esse mecanismo, o índice de acerto subiu para 91,7%.

5.1 Algoritmo adaptado

Como já foi mencionado, o algoritmo sintático de Hobbs original não trata pronomes reflexivos. Neste trabalho, modificamos o segundo passo do algoritmo para incluir a resolução de pronomes reflexivos.

Os passos do algoritmo adaptado são os seguintes:

1. Começar pelo nó *SN*, que imediatamente domina o pronome, e subir na árvore até encontrar um nó *S* ou *SN*. Chamar este nó de *X*, e o caminho percorrido para chegar até ele de *p*;

2. Se o pronome for reflexivo, realizar uma busca em largura, da esquerda para a direita, em todos os filhos de *X* que estejam à esquerda do caminho *p*. Retorne como antecedente o primeiro nó *SN* encontrado;

3. Se o pronome não for reflexivo, realizar uma busca em largura, da esquerda para a direita, em todos os filhos de *X* que estejam à esquerda do caminho *p*. Retorne como antecedente o primeiro nó *SN* encontrado que seja compatível com o pronome, e que contenha um nó *SN* ou *S* entre ele e *X*;

4. Enquanto *X* não for o nó *S* mais alto da sentença:

   (a) A partir do nó *X*, subir na árvore até encontrar um nó *S* ou *SN*. Chamar este nó de *X*, e o caminho percorrido para chegar até ele de *p*;

   (b) Se *X* for um nó *SN* compatível com o pronome, e se o caminho *p* não passar pelo nó *N* que *X* imediatamente domina, retorne *X*;

   (c) Realizar uma busca em largura, da esquerda para a direita, em todos os filhos de *X* que estejam à esquerda do caminho *p*, e retorne o primeiro nó *SN* encontrado que seja compatível com o pronome;

(d) Se X for um nó S, realizar uma busca em largura, da esquerda para a direita, em todas as arestas de X que estejam à direita do caminho p, mas não ir para baixo de nenhum nó SN ou S encontrado. Retorne o primeiro nó SN encontrado que seja compatível com o pronome;

5. Percorrer a árvore das sentenças anteriores do texto, começando pelas sentenças mais próximas. Em cada árvore é realizada uma busca em largura, da esquerda para a direita. Retorne o primeiro nó SN encontrado que seja compatível com o pronome.

O tratamento de pronomes reflexivos é realizado no passo 2. Isso permite que o algoritmo resolva casos como o da sentença (27), a seguir:

(27) O goleiro se machucou.

A Figura 5.5 mostra o funcionamento do algoritmo para esse exemplo:

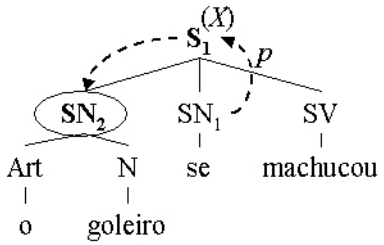


Figura 5.5: Exemplo de resolução de pronome reflexivo.

De acordo com o passo 1, a execução começa no nó SN_1, que imediatamente domina o pronome reflexivo, e sobe para S_1. De acordo com o passo 2, é realizada uma busca em largura nos filhos de S_1, e o nó SN_2 é encontrado e aceito como antecedente.

Como já foi mencionado, o algoritmo original prevê o uso de um tratamento especial para pronomes que aparecem em diálogos; porém, neste trabalho, nenhum tratamento desse tipo foi implementado.

Por outro lado, após uma análise nos corpora, decidimos que a heurística denominada de "pronomes repetidos", mencionada na página 28, deve ser aplicada apenas aos pronomes não reflexivos. Os resultados obtidos são apresentados no Capítulo 6. A sentença (28), extraída do corpus jornalístico, mostra um caso em que essa heurística funciona:

(28) "Nos últimos dias, petebistas que conversaram com **Jefferson**$_i$ alimentaram a versão de que **ele**$_i$ possuiria gravações comprometedoras de aliados e ministros. À Folha, **ele**$_i$ afirmou não ter provas".

No algoritmo adaptado, o número de sentenças anteriores investigadas está limitado a no máximo 5 sentenças, além da sentença que contém a referência pronominal. Esse limite foi definido após uma análise do resultado das execuções do algoritmo para os corpora utilizados. Percebemos que uma busca além das 5 sentenças anteriores é um esforço desnecessário que não contribui para a melhora do desempenho do algoritmo.

No algoritmo original para o inglês, Hobbs menciona como devem ser tratados alguns casos de uso do pronome "*it*". Considerando que não existem casos equivalentes no português, essas situações foram ignoradas.

O mecanismo de restrições semânticas proposto no algoritmo original não foi implementado neste trabalho, pois a proposta era avaliar o algoritmo utilizando apenas informações sintáticas.

5.2 Casos problemáticos

Nesta seção, apresentamos alguns casos em que o algoritmo proposto falha na escolha do referente.

5.2.1 Método de busca utilizado

Considere a sentença (29), a seguir, extraída do corpus jornalístico:

(29) "Compreende-se por que **a ministra**$_j$ se recusa a falar. Nas últimas semanas, **ela**$_j$ assistiu a órgãos subordinados à sua pasta serem alvo de denúncias que causariam perplexidade e horror a **qualquer brasileiro**$_i$, ainda que **ele**$_i$ não fosse, como é o caso de Marina, uma respeitada ambientalista, comprometida por seu trabalho e por sua biografia com a defesa do meio ambiente".

Na resolução do pronome "ela", o algoritmo escolhe corretamente "a ministra" como antecedente. A Figura 5.6 mostra uma parte da árvore sintática para a sentença (29). Como a busca pelo referente é realizada em largura, o algoritmo escolhe "alvo de denúncias que causariam perplexidade e horror a qualquer brasileiro", SN_2, como antecedente do pronome "ele", visto que ele concorda em gênero e número com o pronome.

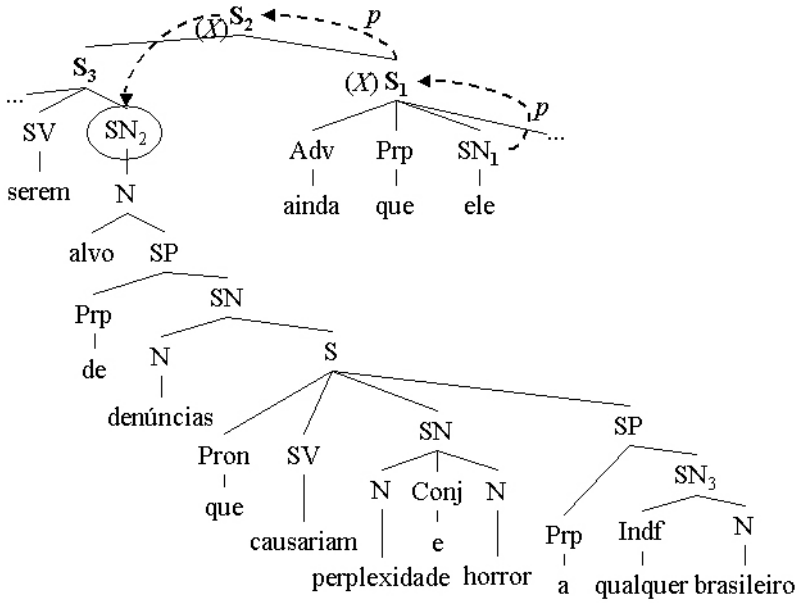


Figura 5.6: Parte da árvore sintática da sentença (29).

De acordo com o passo 1, a execução começa pelo nó SN_1, sobe para o nó S_1, e o caminho percorrido é chamado de p. Como o pronome não é reflexivo, o passo 2 não se aplica. De acordo com o passo 3, é realizada uma busca em largura nos filhos de S_1, mas nenhum nó SN é encontrado à esquerda do caminho p. De acordo com o passo 4.a, o nó S_2 é visitado. Esse nó é chamado de X, e o caminho, de p. O passo 4.b não se aplica, porque S_2 não é um nó SN. De acordo com o passo 4.c, é realizada uma busca em largura, e o nó SN_2 é encontrado e erroneamente aceito como referente.

Se o algoritmo possuísse uma restrição semântica, poderia identificar que o pronome "ele" se refere a um ser humnano. Portanto, o nó SN_2, "alvo de denúncias" poderia ser rejeitado. Com isso, o nó SN_3 seria escolhido.

Na sentença (30), a seguir, o pronome "ela" se refere ao sujeito, "Dorothy". Porém, "a floresta" é erroneamente escolhida como antecedente:

(30) **Dorothy**$_i$ dizia que os deputados desmatavam a floresta que **ela**$_i$ preservava.

A Figura 5.7 mostra a resolução do pronome "ela" na sentença (30).

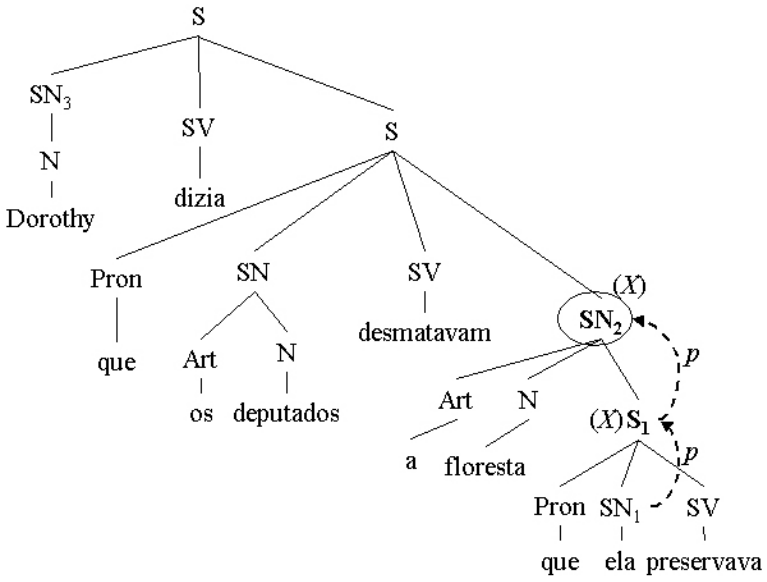


Figura 5.7: Árvore sintática da sentença (30).

De acordo com o passo 1, a execução começa no nó SN_1, e sobe para S_1. Como o pronome não é reflexivo, o passo 2 não se aplica. De acordo com o passo 3, uma busca em largura é realizada nos filhos de S_1, mas nenhum SN é encontrado. Em seguida, como S_1 não é o nó mais alto da sentença, de acordo com o passo 4.a, o nó SN_2 é visitado. De acordo com o passo 4.b, o nó SN_2, isto é, "a floresta" é erroneamente aceito como antecedente do pronome.

Se o algoritmo possuísse uma restrição semântica para a ação que aparece logo após a referência, ou seja, "preservar", o nó SN_2 seria rejeitado, pois "a floresta" não pode ser aceita como agente desta ação. Com isso, o nó SN_3 seria escolhido.

5.2.2 Catáfora

A sentença (31), a seguir, foi extraída do corpus jornalístico.

(31) "A partir de 1960, tornou-**se** necessária a organização de estratégias de desenvolvimento locais".

O pronome destacado corresponde a uma catáfora, e a Figura 5.8 mostra a execução do algoritmo para esse caso.

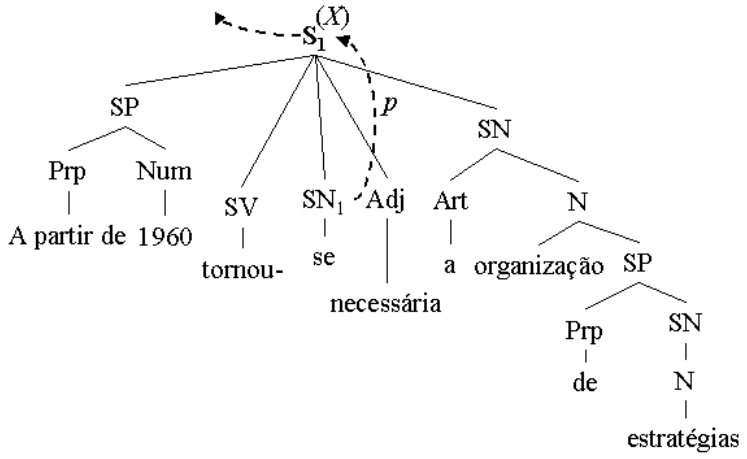


Figura 5.8: Resolução de catáfora no corpus jornalístico.

De acordo com o passo 1, a execução começa pelo nó SN_1, sobe para o nó S_1, e o caminho percorrido é chamado de p. De acordo com o passo 2, é realizada uma busca em largura nos filhos de S_1, à esquerda do caminho p, mas nenhum nó SN é encontrado. Como o pronome é reflexivo, o passo 3 não se aplica. Como o nó S_1 é o mais alto da sentença, o passo 4 não se aplica. De acordo com o passo 5, a busca passa para as sentenças anteriores e um referente incorreto é escolhido.

Na língua portuguesa, a ordem de palavras mais comum é Sujeito-Verbo; porém, no caso da sentença (31), as palavras estão na ordem "inversa" Verbo-Sujeito [18]. Se as palavras da sentença fossem todas colocadas na ordem Sujeito-Verbo-Objeto em um pré-processamento do corpus, antes da execução do algoritmo, o referente correto seria escolhido.

No corpus Summ-it existe apenas um caso de catáfora, que está destacado na sentença (32). Como a catáfora aparece logo no início do texto, não há sentenças anteriores a serem investigadas:

(32) "**Eles**$_i$ não acharam nenhum tesouro de pirata, nem esmeraldas, rubis ou moedas de ouro e prata. (...). Mas **a equipe internacional de arqueólogos subaquáticos**$_i$, incluindo brasileiros, que trabalhou na foz do rio Arade, no sul de Portugal, já pôde pelo menos sepultar um mito - o de que ali haveria um navio viking naufragado".

A Figura 5.9 mostra a execução do algoritmo para um fragmento da sentença (32).

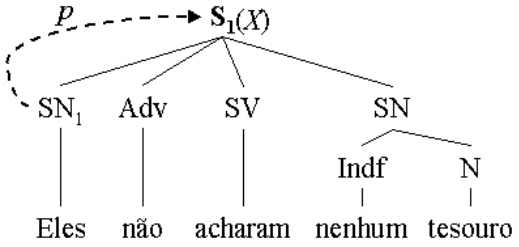


Figura 5.9: Resolução de catáfora no corpus Summ-it.

De acordo com o passo 1, a execução começa pelo nó SN_1 e sobe para o nó S_1. O nó S_1 é chamado de X, e o caminho percorrido, de p. Como o pronome não é reflexivo, o passo 2 não se aplica. De acordo com o passo 3, é realizada uma busca em largura nos filhos de X à esquerda de p, mas não existem arestas à esquerda do caminho p. Como o nó S_1 é o mais alto da sentença, o passo 4 não se aplica. De acordo com o passo 5, a busca passa para as sentenças anteriores. Porém, como não existem sentenças anteriores, a execução falha.

Neste caso, o algoritmo teria acertado se possuísse uma condição para no caso de a referência ser a primeira palavra do texto, realizar uma busca nas próximas sentenças e selecionar o primeiro sujeito que for compatível com o pronome.

Considere a sentença (33), a seguir:

(33) "Os altos funcionários que **lhe** ouviam compreenderam a importância.".

A Figura 5.10 mostra a execução do algoritmo para a resolução do pronome "lhe". Nota-se que o algoritmo escolhe um sintagma nominal que está depois do pronome, o que poderia caracterizar uma catáfora. Porém, o referente está na sentença anterior, o que caracteriza uma anáfora inter-sentencial. A esse tipo de erro denominamos "catáfora falsa".

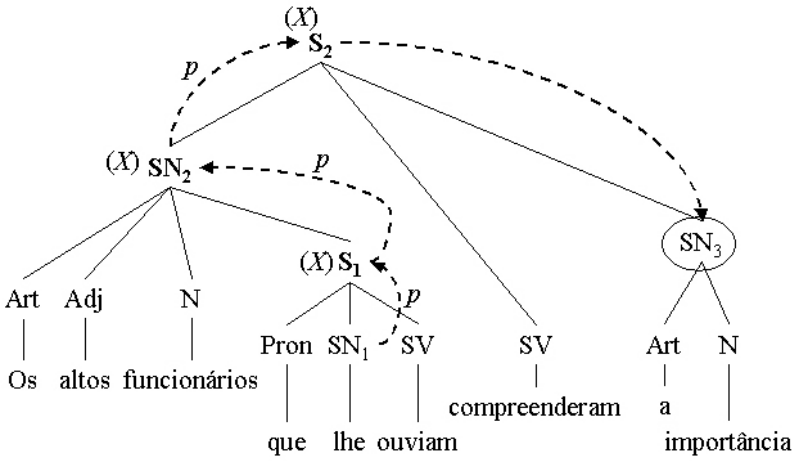


Figura 5.10: Resolução do pronome na sentença (33).

De acordo com o passo 1, a execução começa pelo nó SN_1 e sobe para o nó S_1. O nó S_1 é chamado de X, e o caminho percorrido, de p. Como o pronome não é reflexivo, o passo 2 não se aplica. De acordo com o passo 3, é realizada uma busca em largura nos filhos de X à esquerda de p, mas nenhum nó SN é encontrado. Como S_1 não é o nó mais alto da sentença, de acordo com o passo 4.a o nó SN_2 é visitado. De acordo com o passo 4.b, o nó SN_2 não é aceito como antecedente do pronome, porque o número de SN_2 (plural) não é compatível com o número de SN_1 (singular). Em seguida, de acordo com o passo 4.c, é realizada uma busca em largura nos filhos de SN_2, mas não há nó SN à esquerda do caminho p. De acordo com o passo 4.d, é realizada uma busca nos filhos de SN_2 à direita do caminho p, mas não existem arestas à direita do caminho p. Como o nó SN_2 não é o mais alto da sentença, de acordo com o passo 4.a, o nó S_2 é visitado. O passo 4.b não se aplica, porque S_2 não é um nó SN. Em seguida, de acordo com o passo 4.c, é realizada uma busca em largura nos filhos de S_2, mas não existem arestas à esquerda do caminho p. De acordo com o passo 4.d, é realizada uma busca nos filhos de S_2 à direita do caminho p. O nó SN_3 é encontrado e erroneamente aceito como referente do pronome.

Nenhum dos casos de catáfora presentes nos corpora utilizados foi resolvido pelo passo 4.d. Se este passo for retirado do algoritmo, o seu desempenho poderia ser melhor.

5.2.3 Elipse

Considere a sentença (34), a seguir:

(34) **As filhas de Alzino**$_i$ foram para Palma de Mallorca há sete anos. Já construíram quatro casas no bairro. **Elas**$_i$ não tinham emprego em Uruaçu.

A Figura 5.11 mostra a execução do algoritmo para resolução do pronome “Elas”.

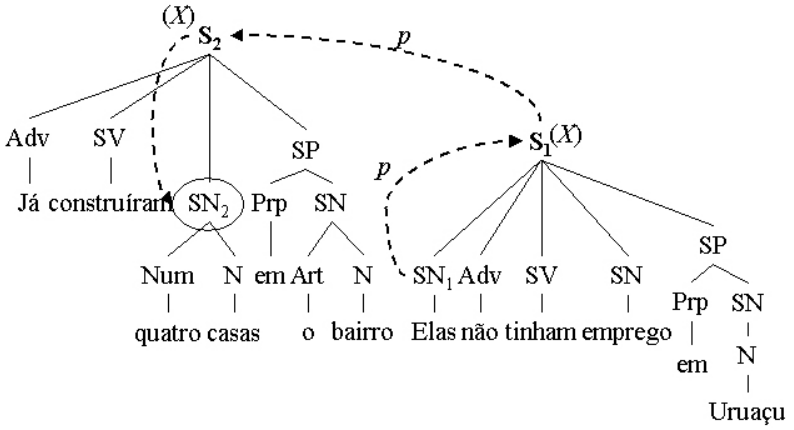


Figura 5.11: Resolução do pronome na sentença (34).

De acordo com o passo 1, a execução começa pelo nó SN_1, sobe para o nó S_1, e o caminho percorrido é chamado de p. Como o pronome não é reflexivo, o passo 2 não se aplica. De acordo com o passo 3, é realizada uma busca em largura nos filhos de S_1, mas nenhum nó SN é encontrado à esquerda do caminho p. Como S_1 é o nó mais alto da sentença, o passo 4 não se aplica. De acordo com o passo 5, o nó S_2 é visitado. É realizada uma busca em largura, da esquerda para a direita, nos filhos de S_2, e o nó SN_2, "quatro casas", é erroneamente aceito como antecedente do pronome "Elas".

Um aspecto importante do algoritmo original de Hobbs é que ele supõe a utilização de um pré-processamento anterior à resolução das referências, cuja função é recuperar elementos omitidos (elipses) utilizando informações sintáticas.

Neste trabalho, nenhum mecanismo de resolução de elipses foi utilizado; porém, existem alguns trabalhos, como o de Maduro [26], que poderiam ser utilizados para esse fim.

Na sentença (34) existe uma elipse. Se essa elipse tivesse sido resolvida antes do processamento, o algoritmo teria escolhido o antecedente correto. A sentença (35) mostra como ficaria a sentença após a resolução da elipse:

(35) **As filhas de Alzino**$_i$ foram para Palma de Mallorca há sete anos. ***As filhas de Alzino***$_i$ já construíram quatro casas no bairro. **Elas**$_i$ não tinham emprego em Uruaçu.

A Figura 5.12 mostra a resolução do pronome em SN_1 se a elipse tivesse sido resolvida.

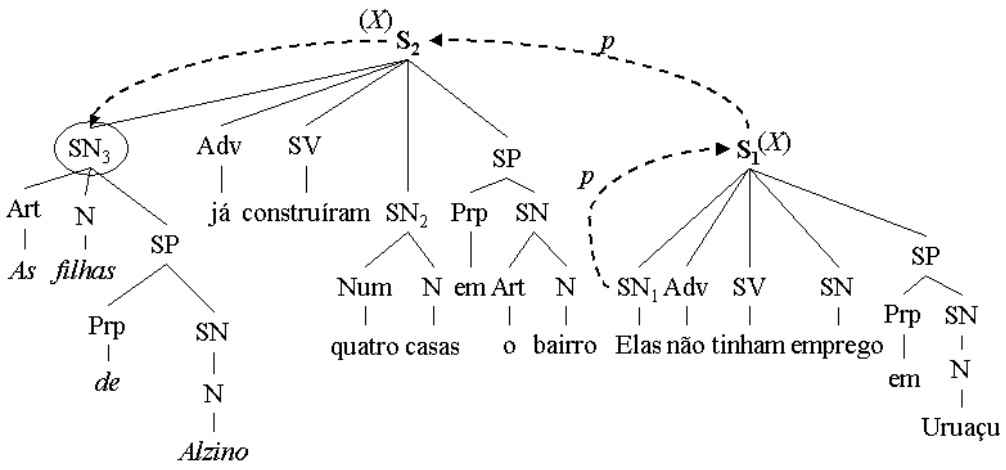


Figura 5.12: Resolução do pronome na sentença (35).

De acordo com o passo 1, a execução começa pelo nó SN_1, sobe para o nó S_1, e o caminho percorrido é chamado de p. Como o pronome não é reflexivo, o passo 2 não se aplica. De acordo com o passo 3, é realizada uma busca em largura nos filhos de S_1, mas nenhum nó SN é encontrado à esquerda do caminho p. Como S_1 é o nó mais alto da sentença, o passo 4 não se aplica. De acordo com o passo 5, o nó S_2 é visitado. É realizada uma busca em largura, da esquerda para a direita, nos filhos de S_2, e o nó SN_3 é aceito como antencedente do pronome "Elas".

5.2.4 Sentenças que são referentes

O algoritmo de Hobbs não é capaz de lidar com referências cujo referente é a própria sentença; por isso, nas ocorrências desse tipo, o algoritmo falha. A sentença (36), extraída do corpus jornalístico, é um exemplo desse caso. A referência pronominal está destacada em negrito:

(36) "No que se refere à **devastação causada pela corrupção na Amazônia**$_i$, o governo Lula não pode dizer que não teve chance de, ao menos, contribuir para reduzi-**la**$_i$ drasticamente. Poderia tê-**lo** feito por meio de uma assinatura".

O pronome "la" se refere a "devastação causada pela corrupção na Amazônia". O pronome "lo" se refere a "contribuir para reduzir a devastação causada pela corrupção da amazônia".

A sentença (37), extraída do corpus literário, é mais um exemplo de referência à própria sentença:

(37) "Poderia convidar alguns de vós em comissão dos outros a vir ver comigo os loucos reclusos; mas não **o** faço, porque seria dar-vos razão do meu sistema, o que não farei a leigos nem a rebeldes".

O pronome "o" se refere a "convidar alguns de vós em comissão dos outros a vir ver comigo os loucos reclusos".

A sentença (38) é mais um exemplo de referência a uma sentença inteira, desta vez extraída do corpus jurídico:

(38) "Nos termos do artigo 6º, a Ordem dos Médicos tem por finalidades, entre outras, 'fomentar e defender os interesses da profissão médica a todos os níveis, nomeadamente no respeitante à promoção sócio-profissional, à segurança social e às relações de trabalho' - alínea b) e 'dar parecer sobre todos os assuntos relacionados com o ensino, com o exercício da medicina e com a organização dos serviços que se ocupem da saúde, sempre que julgue conveniente fazê-**lo**, junto das entidades oficiais competentes ou quando por estes for consultada' - alínea b)".

O pronome "lo" se refere a "dar parecer sobre todos os assuntos relacionados com o ensino, com o exercício da medicina e com a organização dos serviços que se ocupem da saúde".

A sentença (39), extraída do corpus Summ-it, é um exemplo de referência à própria sentença:

(39) "A idéia, iniciada em 1976, era coletar e preservar criogenicamente (em baixas temperaturas) amostras celulares de **animais ameaçados de extinção**$_i$, com a esperança de estudá-los$_i$ e, quem sabe, ressuscitá-los quando a tecnologia assim **o** permitisse".

O pronome "o" se refere a "ressuscitar animais ameaçados de extinção".

5.2.5 Pronomes repetidos

Existem alguns casos em que a heurística "pronomes repetidos", proposta por Hobbs, compromete o desempenho do algoritmo. A sentença (40), extraída do corpus literário, contém um exemplo de pronomes repetidos que não são correferentes.

(40) "**Um dos tios dele**$_i$, caçador de pacas perante o Eterno, e não menos franco, admirou-**se**$_i$ de semelhante escolha e disse-**lho**$_j$. **Simão Bacamarte**$_j$ explicou-**lhe**$_i$ que D. Evarista reunia condições fisiológicas e anatômicas de primeira ordem, digeria com facilidade, dormia regularmente, tinha bom pulso, e excelente vista; estava assim apta para dar-**lhe**$_j$ filhos robustos, sãos e inteligentes".

Na resolução do pronome reflexivo "se", o sintagma "Um dos tios dele" é corretamente escolhido como referente. Já na primeira ocorrência do pronome "lhe", na contração "lho" ("lhe"+"o"), o algoritmo escolhe erroneamente "Um dos tios dele" como antecedente. Na sentença seguinte há a segunda ocorrência do pronome "lhe". O algoritmo supõe esta ocorrência como um pronome repetido e escolhe "Um dos tios dele" como antecedente novamente. O mesmo acontece com a terceira ocorrência do pronome "lhe".

Finalmente, existem casos em que não existe um sintagma nominal compatível em gênero e número com a anáfora, como na sentença (41), que foi extraída do corpus literário:

(41) "(...) mas Simão Bacamarte não afrouxava; ia de rua em rua, de casa em casa, espreitando, interrogando, estudando; e quando colhia um enfermo levava-**o** com a mesma alegria com que outrora **os** arrebanhava às dúzias".

Não existe um sintagma nominal no plural que seja compatível com o pronome "os", e, por isso, o algoritmo falha. Entretanto, na resolução do pronome "o", o sintagma "um enfermo" é corretamente escolhido como referente. A Figura 5.13 mostra a execução do algoritmo.

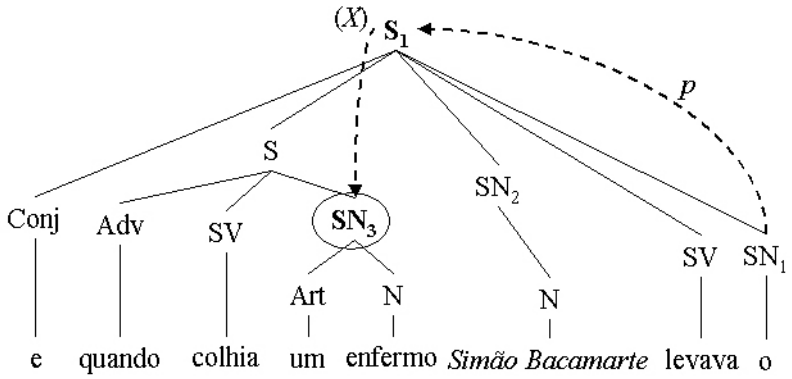


Figura 5.13: Resolução do pronome na sentença (41).

De acordo com o passo 1, a execução começa pelo nó SN_1 e sobe para o nó S_1. O nó S_1 é chamado de X, e o caminho percorrido, de p. Como o pronome não é reflexivo, o passo 2 não se aplica. De acordo com o passo 3, é realizada uma busca em largura nos filhos de X à esquerda de p, e o nó SN_2 é encontrado. Porém, ainda de acordo com o passo 3, como o nó SN_2 não possui um outro nó S ou SN entre ele e X, ele é descartado. A busca em largura prossegue, e o nó SN_3 é escolhido e aceito como referente.

O algoritmo poderia acertar se possuísse um mecanismo que identificasse que as duas referências pronominais estão relacionadas com ações de significados muito próximos, "levar" e "arrebanhar". Dessa forma, poderia considerar que as referências possuem o mesmo referente.

A sentença (42), que foi extraída do corpus literário, contém outro exemplo de anáfora plural:

(42) "(...) abria os braços e alargava as pernas para dar-**lhes** certa feição de raios".

O pronome "lhes" se refere a "os braços" e "as pernas"; porém, o algoritmo seleciona somente "os braços" como antecedente.

O algoritmo poderia acertar se identificasse que as duas sentenças conectadas pela conjunção "e" possuem verbos com significados próximos, "abrir" e "alargar" e, assim, como o pronome está no plural, selecionar os objetos das duas sentenças como referentes.

A Figura 5.14 mostra a execução do algoritmo para a sentença (42).

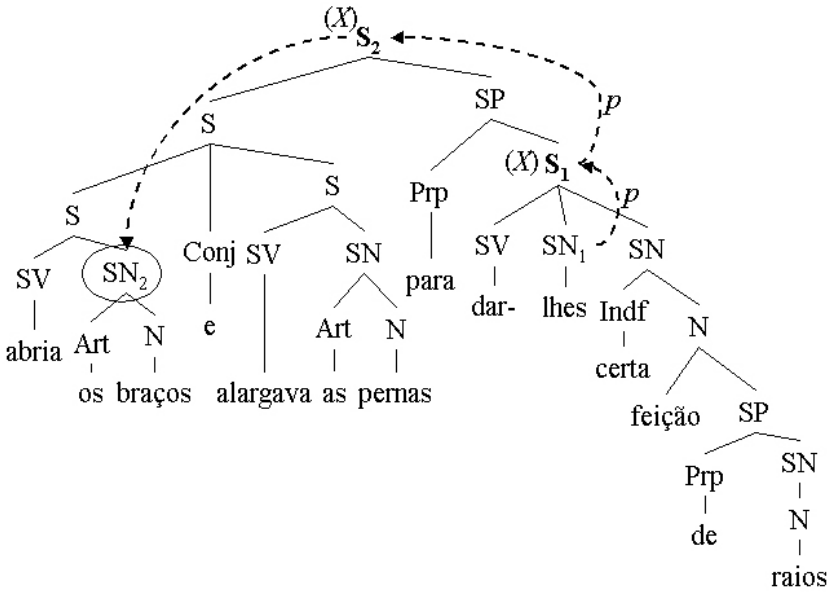


Figura 5.14: Resolução de anáfora no plural.

De acordo com o passo 1, a execução começa pelo nó SN_1, sobe para o nó S_1, e o caminho percorrido é chamado de p. Como o pronome não é reflexivo, o passo 2 não se aplica. De acordo com o passo 3, é realizada uma busca em largura nos filhos de S_1, mas nenhum nó SN é encontrado à esquerda do caminho p. De acordo com o passo 4.a, o nó S_2 é visitado. Esse nó é chamado de X, e o caminho percorrido, de p. O passo 4.b não se aplica, porque S_2 não é um nó SN. De acordo com o passo 4.c, é realizada uma busca em largura, e o nó SN_2 é encontrado e erroneamente aceito como referente.

5.3 Considerações

Nesse capítulo foi descrito o algoritmo original de Hobbs, assim como sua adaptação para o português proposta neste trabalho.

Também foram mostrados os casos em que o algoritmo não consegue encontrar o referente correto, isto é, falha. Alguns desses casos são de referência a uma sentença inteira ou a parte de uma sentença, como nas sentenças (38) e (39). Outros casos são os de referência a mais de um sintagma nominal, como na sentença (42), além de casos como o da sentença (34), que possui uma elipse não resolvida.

O algoritmo de Hobbs não conseguiu resolver corretamente nenhum dos casos de catáfora, como na sentença (31).

No próximo capítulo, é descrito o processo de avaliação utilizado neste trabalho, e são apresentados os resultados obtidos com o algoritmo de Hobbs adaptado para o português.

Capítulo 6

Avaliação do Algoritmo

Neste capítulo, descrevemos o processo de avaliação do algoritmo e apresentamos os resultados.

Conforme afirmado no capítulo 4, após algumas correções nas marcações dos corpora, a quantidade de anáforas marcadas foi reduzida; por isso os resultados para o algoritmo de Lappin e Leass apresentados a seguir estão ligeiramente diferentes dos que constam em Coelho [11].

6.1 Corpus jornalístico

No corpus jornalístico, consideramos que o algoritmo encontrou um referente correto se a solução gerada for igual a solução marcada, ou se a solução gerada for correferente da solução marcada.

Na sentença (43), por exemplo, quando o algoritmo tenta resolver o pronome "eles", consideramos que o algoritmo acertou se "gorilas e chimpanzés" for escolhido como referente. Se o algoritmo escolher "fins alimentícios", por exemplo, consideramos que o algoritmo errou. Quando o algoritmo tenta resolver o pronome "los", consideramos que o algoritmo acertou apenas se ele escolher o referente "gorilas e chimpanzés". Se o algoritmo escolher o pronome "eles" como referente do pronome "los", consideramos que o algoritmo acertou se o referente escolhido para o pronome "eles" for "gorilas e chimpanzés".

(43) "A mortandade dos **gorilas e chimpanzés**$_i$, ainda por cima para fins alimentícios, tem também um significado simbólico de assustar . Como **eles**$_i$ têm o código genético quase igual ao dos seres humanos e são fisicamente semelhantes a nós, comê-**los**$_i$ parece um ato de antropofagia".

A Tabela 6.1 contém uma comparação das referências pronominais resolvidas corretamente pelo algoritmo de Hobbs, usado neste trabalho, e pelo algoritmo de Lappin e Leass,

adaptado por Coelho [11]. As porcentagens entre parênteses têm como base os valores da coluna "Quantidade".

Tabela 6.1: Resultado dos dois algoritmos no corpus jornalístico por tipo de pronome.

Pronomes	Quantidade	Algoritmo de Hobbs	Algoritmo de Lappin e Leass
Reflexivos	81	59 (72,84%)	32 (39,51%)
Não reflexivos	90	47 (52,22%)	47 (52,22%)
Total	171	106 (61,99%)	79 (46,20%)

O algoritmo de Hobbs obteve melhor resultado na resolução de pronomes reflexivos graças a adaptação realizada no passo 2, isto é, o algoritmo dá preferência aos candidatos que estejam no mesmo nível que a referência na árvore. Dos 59 pronomes reflexivos que o algoritmo selecionou o antecedente correto, 42 foram resolvidos no passo 2.

O algoritmo de Lappin e Leass [11] também possui uma adaptação para pronomes reflexivos que seleciona apenas candidatos intra-sentenciais para fazer a resolução.

Curiosamente, o algoritmo de Lappin e Leass e o algoritmo de Hobbs tiveram o mesmo resultado na resolução de pronomes não reflexivos.

Dentre os 171 pronomes resolvidos nesse corpus, os algoritmos coincidiram na escolha do mesmo sintagma nominal como antecedente para 83 pronomes (48,53%).

Dentre os 106 pronomes que o algoritmo de Hobbs resolveu corretamente, 67 pronomes também foram resolvidos corretamente pelo algoritmo de Lappin e Leass.

A Tabela 6.2 mostra a quantidade de referências por tipo de anáfora resolvida corretamente pelo algoritmo de Hobbs e pelo algoritmo de Lappin e Leass. Apesar do limite de 5 sentenças anteriores a serem investigadas, nenhuma resolução de pronome nesse corpus chegou a atingir esse limite. O número máximo de sentenças anteriores pesquisadas pelo algoritmo de Hobbs nesse corpus foi dois.

Tabela 6.2: Resultado no corpus jornalístico por tipo de referência.

Referências	Quantidade	Algoritmo de Hobbs	Algoritmo de Lappin e Leass
Intra-sentenciais	109	75 (68,81%)	49 (44,95%)
Inter-sentenciais	62	31 (50,00%)	30 (48,39%)
Total	171	106 (61,99%)	79 (46,20%)

No corpus jornalístico existem 3 casos de catáfora, um deles corresponde ao exemplo da Figura 5.8 (veja página 34), sentença (31).

Como já foi mencionado no Capítulo 5, esse algoritmo não é capaz de lidar com referência cujo referente seja a própria sentença; por isso, nas 3 ocorrências desse tipo no corpus jornalístico, o algoritmo falhou. A sentença (36), repetida aqui por conveniência, contém um desse casos:

(44) "No que se refere à **devastação causada pela corrupção na Amazônia**$_i$, o governo Lula não pode dizer que não teve chance de, ao menos, contribuir para reduzi-la$_i$ drasticamente. Poderia tê-**lo** feito por meio de uma assinatura".

No corpus jornalístico, não existem casos de "catáfora falsa".

A heurística "pronomes repetidos", descrita no Capítulo 5, foi utilizada em 5 casos no corpus jornalístico. Em todos os casos os pronomes eram correferentes. A sentença (28) (repetida em (45) por conveniência) apresentou um desses casos.

(45) "Nos últimos dias, petebistas que conversaram com **Jefferson**$_i$ alimentaram a versão de que **ele**$_i$ possuiria gravações comprometedoras de aliados e ministros. À Folha, **ele**$_i$ afirmou não ter provas".

6.2 Corpus literário

No corpus literário, os resultados foram avaliados da mesma forma que no corpus jornalístico.

Visando tornar o resultado independente da ferramenta de marcação adotada, ajustamos os gêneros dos personagens do texto manualmente.

Na Tabela 6.3 é feita a comparação dos resultados do algoritmo de Hobbs com os do algoritmo de Lappin e Leass por tipo de pronome.

Tabela 6.3: Resultado dos dois algoritmos no corpus literário por tipo de pronome.

Pronomes	Quantidade	Algoritmo de Hobbs	Algoritmo de Lappin e Leass
Reflexivos	122	70(57,38%)	49 (40,16%)
Não reflexivos	468	191 (40,81%)	164 (35,04%)
Total	590	261 (44,24%)	213 (36,10%)

Novamente, o algoritmo de Hobbs obteve mais acertos na resolução de pronomes reflexivos. Dos 70 pronomes reflexivos resolvidos corretamente, 39 foram resolvidos no passo 2. Quanto a resolução de pronomes não reflexivos, o algoritmo de Hobbs também teve um desempenho melhor.

Dentre os 590 pronomes resolvidos nesse corpus, os algoritmos coincidiram na escolha do mesmo sintagma nominal como antecedente para 222 pronomes (37,63%).

Dentre os 261 pronomes que o algoritmo de Hobbs resolveu corretamente, 149 foram resolvidos corretamente pelo algoritmo de Lappin e Leass também.

A Tabela 6.4 contém a taxa de acertos dos mesmos algoritmos por tipo de anáfora. Nas anáforas inter-sentenciais, utilizou-se o limite máximo de 5 sentenças anteriores a serem investigadas para evitar esforço desnecessário. Na resolução dos pronomes desse

Tabela 6.4: Resultado no corpus literário por tipo de referência.

Referências	Quantidade	Algoritmo de Hobbs	Algoritmo de Lappin e Leass
Intra-sentenciais	248	141 (56,85%)	99 (39,92%)
Inter-sentenciais	342	120 (35,09%)	114 (33,33%)
Total	590	261 (44,24%)	213 (36,10%)

corpus, houve 6 casos em que a execução do algoritmo atingiu esse limite, tendo falhado na busca pelo referente.

No corpus literário, existem 11 casos de catáfora e 10 casos em que o referente é uma sentença inteira. Em todos eles o algoritmo falhou. A sentença (37), repetida em (46), mostrou um desses casos. Também existem 6 erros de "catáfora falsa". Um desses casos foi mostrado na Figura 5.10, na página 36.

(46) "Poderia convidar alguns de vós em comissão dos outros a vir ver comigo os loucos reclusos; mas não **o** faço, porque seria dar-vos razão do meu sistema, o que não farei a leigos nem a rebeldes".

A heurística "pronomes repetidos" foi utilizada em 71 casos; em apenas 23 casos os pronomes não eram correferentes. A sentença (40) (repetida em (47) por conveniência) contém um desses casos.

(47) "**Um dos tios dele**$_i$, caçador de pacas perante o Eterno, e não menos franco, admirou-**se**$_i$ de semelhante escolha e disse-**lho**$_j$. **Simão Bacamarte**$_j$ explicou-**lhe**$_i$ que D. Evarista reunia condições fisiológicas e anatômicas de primeira ordem, digeria com facilidade, dormia regularmente, tinha bom pulso, e excelente vista; estava assim apta para dar-**lhe**$_j$ filhos robustos, sãos e inteligentes".

6.3 Corpus jurídico

Nesse corpus, a anotação das referências pronominais não incluiu os pronomes reflexivos.

No corpus jurídico, consideramos que o algoritmo encontrou um referente correto somente se a solução gerada for igual a solução marcada.

Na Tabela 6.5 são comparados os resultados obtidos com os algoritmos nesse corpus.

Tabela 6.5: Resultado dos dois algoritmos no corpus jurídico.

Pronomes	Quantidade	Algoritmo de Hobbs	Algoritmo de Lappin e Leass
Não reflexivos	287	124 (43,21%)	98 (34,15%)

A Tabela 6.6 contém o resultado do algoritmo de Hobbs por tipo de referência. O número máximo de sentenças anteriores investigadas na resolução dos pronomes desse corpus foi 2, apesar do limite de 5 sentenças.

Tabela 6.6: Resultado no corpus jurídico por tipo de referência.

Referências	Quantidade	Algoritmo de Hobbs	Algoritmo de Lappin e Leass
Intra-sentenciais	218	101 (46,33%)	69 (31,65%)
Inter-sentenciais	69	23 (33,33%)	29 (42,03%)
Total	287	124 (43,21%)	98 (34,15%)

Na primeira tabela, é possível observar que o algoritmo de Hobbs obteve um resultado melhor que o de Lappin e Leass para esse corpus.

Dentre os 287 pronomes resolvidos nesse corpus, os algoritmos coincidiram na escolha do mesmo sintagma nominal como antecedente para 100 pronomes (34,84%).

Dentre os 124 pronomes que o algoritmo de Hobbs resolveu corretamente, 59 também foram resolvidos corretamente pelo algoritmo de Lappin e Leass.

Na segunda tabela é possível observar que o algoritmo de Hobbs teve um desempenho melhor na resolução de referências intra-sentenciais e um desempenho pior na resolução de referências inter-sentenciais.

A heurística "pronomes repetidos" foi utilizada em 23 casos e em apenas 4 casos os pronomes não eram correferentes.

No corpus jurídico não existem catáforas, mas existe uma exófora, 8 casos de "catáfora falsa" e 3 casos em que o referente é uma sentença. A sentença (38) (veja página 39) mostra um caso em que o pronome referencia um sentença inteira.

6.4 Corpus Summ-it

No corpus Summ-it, consideramos que o algoritmo encontrou um referente correto somente se a solução gerada pertencer à cadeia de correferência a que pertence o pronome, conforme marcação descrita na seção 4.4.

O Summ-it possui 143 referências de pronomes pessoais marcadas. A marcação das referências pronominais inclui apenas 1 pronome reflexivo.

Na Tabela 6.7, é apresentado o resultado obtido nesse corpus.

Tabela 6.7: Resultado do algoritmos de Hobbs no corpus Summ-it por tipo de pronome.

Pronomes	Quantidade	Algoritmo de Hobbs	Algoritmo de Lappin e Leass
Reflexivos	1	1(100%)	1 (100%)
Não reflexivos	142	78 (54,93%)	72 (50,70%)
Total	143	79 (55,24%)	73 (51,05%)

Ambos os algoritmos tiveram resultados muito próximos. Dentre os 143 pronomes resolvidos nesse corpus, os algoritmos coincidiram na escolha do mesmo sintagma nominal como antecedente para 79 pronomes (55,24%).

Dentre os 79 pronomes que o algoritmo de Hobbs resolveu corretamente, 49 pronomes também foram resolvidos corretamente pelo algoritmo de Lappin e Leass.

A Tabela 6.8 contém os resultados por tipo de referência.

Nesse corpus, houve um caso em que a busca no algoritmo de Hobbs chegou ao limite máximo de sentenças anteriores e falhou. Esse caso correponde à resolução da única ocorrência de catáfora nesse corpus, que foi demonstrada na sentença (32), na página 34.

Nos outros casos, o número máximo de sentenças anteriores investigadas na resolução dos pronomes desse corpus foi 3, apesar do limite de 5 sentenças.

Tabela 6.8: Resultado no corpus Summ-it por tipo de referência.

Referências	Quantidade	Algoritmo de Hobbs	Algoritmo de Lappin e Leass
Intra-sentenciais	59	36 (61,02%)	24 (40,68%)
Inter-sentenciais	84	43 (51,19%)	49 (58,33%)
Total	143	79 (55,24%)	73 (51,05%)

No corpus Summ-it aparecem 3 casos de "catáfora falsa" e um caso em que o referente é a sentença inteira, como foi destacado na sentença (39), repetida em (48) por conveniência.

(48) "A idéia, iniciada em 1976, era coletar e preservar criogenicamente (em baixas temperaturas) amostras celulares de animais ameaçados de extinção, com a esperança de estudá-los e, quem sabe, ressuscitá-los quando a tecnologia assim **o** permitisse".

A heurística "pronomes repetidos" foi utilizada em doze casos; em apenas dois deles os pronomes não eram correferentes.

6.5 Resultado total

Na Tabela 6.9, é apresentado o total dos resultados obtidos para os quatro corpora.

Tabela 6.9: Resultado geral dos algoritmos para todos os 4 corpora.

Pronomes	Quantidade	Algoritmo de Hobbs	Algoritmo de Lappin e Leass
Reflexivos	204	107(52,45%)	63 (30,88%)
Não reflexivos	987	439 (44,48%)	383 (38,80%)
Total	1191	546 (45,84%)	446 (37,45%)

Considerando os resultados dos quatro corpora, o algoritmo de Hobbs obteve um resultado melhor na resolução dos pronomes reflexivos e não reflexivos.

A Tabela 6.10 contém o total dos resultados por tipo de referência.

Considerando o total dos resultados obtidos por tipo de referência, o algoritmo de Hobbs obteve um resultado melhor na resolução das referências intra-sentenciais e o algoritmo de Lappin e Leass obteve um resultado melhor na resolução das referências inter-sentenciais.

Tabela 6.10: Resultado nos corpora por tipo de referência.

Referências	Quantidade	Algoritmo de Hobbs	Algoritmo de Lappin e Leass
Intra-sentenciais	634	332 (52,37%)	225 (35,49%)
Inter-sentenciais	557	214 (38,42%)	221 (39,68%)
Total	1191	546 (45,84%)	446 (37,45%)

6.6 Considerações

Neste capítulo, foi descrito o processo de avaliação do algoritmo de Hobbs.

Os resultados obtidos mostram que o algoritmo de Hobbs teve um desempenho equivalente ao algoritmo de Lappin e Leass na resolução de pronomes não reflexivos em textos com sentenças mais simples, como os corpora jornalístico e Summ-it. Também mostram que a adaptação para a resolução do pronome reflexivo foi bem sucedida e contribuiu para a melhoria do desempenho do algoritmo.

No próximo capítulo, são apresentadas as conclusões e sugestões para trabalhos futuros.

Capítulo 7

Considerações finais

Neste trabalho propusemos e avaliamos uma adaptação do algoritmo de Hobbs para a língua portuguesa. O algoritmo de Hobbs, que já era capaz de resolver referências pronominais intra- e inter-sentenciais, foi estendido para que fosse capaz de resolver pronomes reflexivos. Uma avaliação da proposta foi realizada utilizando quatro corpora distintos. Os resultados foram comparados com os obtidos pela adaptação do algoritmo de Lappin e Leass desenvolvida por Coelho [11].

A adaptação do algoritmo de Hobbs para a resolução de pronomes reflexivos se mostrou eficaz. Os resultados obtidos com a resolução de pronomes reflexivos foram melhores que os do algoritmo de Lappin e Leass.

A heurística denominada "pronomes repetidos" se mostrou eficaz pois, em mais de 70% dos casos em que foi aplicada, os pronomes eram correferentes.

O recurso de resolução de catáfora do algoritmo de Hobbs não se mostrou eficaz. Em nenhum caso de catáfora esse recurso resultou na escolha do referente correto.

Apesar do limite máximo de cinco sentenças anteriores para a busca, poucas vezes esse limite foi alcançado. Esse limite serviu apenas para evitar esforço desnecessário, e em nenhum momento impediu que o algoritmo escolhesse o referente correto.

O fato de não termos utilizado um mecanismo de resolução de elipse afetou o desempenho do algoritmo. No corpus jornalístico, dos casos casos em que o algoritmo falhou na resolução de pronomes reflexivos, 29% foram causados por elipses.

Se compararmos os resultados obtidos nesta avaliação com os resultados obtidos por Chaves [9] na adaptação do algoritmo de Mitkov, verificamos que as taxas de sucesso do algoritmo de Mitkov são superiores nos três corpora utilizados por Chaves: jornalístico, literário e jurídico. Porém, apesar de os corpora serem os mesmos, em nossa avaliação do algoritmo de Hobbs realizamos algumas correções nas saídas geradas pelas ferramentas utilizadas na marcação dos corpora.

Devido às correções realizadas nos corpora, a comparação com os resultados do al-

goritmo de Mitkov não pode ser detalhada como na comparação com os resultados do algoritmo de Lappin e Leass.

Os resultados mostram que a adaptação do algoritmo de Hobbs obteve resultados equivalentes à adaptação do algoritmo de Lappin e Leass em vários casos. Muitas vezes os algoritmos escolheram o mesmo sintagma nominal como antecedente para um pronome, apesar do algoritmo de Hobbs parecer muito simples quando comparado com o algoritmo de Lappin e Leass. Esses resultados ainda são bastante inferiores aos obtidos pelo algoritmo original de Hobbs para o inglês (88,3%). Entretanto, é importante ressaltar que corpora maiores foram utilizados neste trabalho para avaliação. A complexidade das sentenças em português usadas também pode ter afetado o desempenho do algoritmo.

7.1 Trabalhos futuros

Como aperfeiçoamento deste trabalho, pretendemos utilizar um mecanismo resolvedor de elipses, como o proposto por Maduro [26], em um pré-processamento do corpus, antes de executar o algoritmo. Acreditamos que uma melhora significativa do desempenho pode ser obtida recuperando alguns elementos omitidos, que poderiam ser escolhidos durante a busca realizada pelo algoritmo.

Outra possiblidade de expansão seria incluir um mecanismo capaz de manipular um conjunto de restrições semânticas, que funcionariam como um filtro. Dessa forma, qualquer entidade do texto que fosse sugerida como antecedente deveria ser filtrada por esse mecanismo. De acordo com Hobbs, com o uso desse mecanismo, a taxa de acerto subiu para 91,7%.

Referências Bibliográficas

[1] A.M. Aires, J.C.B. Coelho, S. Collovini, P. Quaresma, e R. Vieira. Avaliação de centering em resolução pronominal da língua portuguesa. Em *Taller de Herramientas y Recursos Lingüísticos para el Español y el Portugués*. Iberamia, 2004.

[2] M. de Assis. *O Alienista*. VirtualBooks, 2002.

[3] F.A. Barros e J. Robin. Processamento de linguagem natural. *Revista Eletrônica de Iniciação Científica*, I(II), Novembro 2001.

[4] E. Bick. *The parsing system PALAVRAS: Automatic grammatical analysis of Portuguese in a constraint grammar frammework*. Tese de Doutorado, Århus University, 2000.

[5] S.E. Brennan, M.W. Friedman, e C.J. Pollard. A centering approach to pronouns. Em *Proceedings of the 25th Annual Meeting on Association for Computational Linguistics*, páginas 155–162. Morristown, NJ, EUA, 1987.

[6] J.G. Carbonell e R.D. Brown. Anaphora resolution: A multi-strategy approach. Em *Proceedings of the 12th International Conference on Computational Linguistics (COLING'88)*, páginas 96–101. Blaisdell, Hawai, 1988.

[7] D.P. Cegalla. *Nova minigramática da língua portuguesa*. Companhia Editora Nacional, São Paulo, 1ª edição, 2004.

[8] E. Charniak. *Toward a model of children's story comprehension*. MIT, Massachusetts, 1972.

[9] A.R. Chaves. A resolução de anáforas pronominais da língua portuguesa com base no algoritmo de Mitkov. Dissertação de Mestrado, Universidade Federal de São Carlos, 2007.

[10] N. Chomsky. Remarks on nominalization. Em *Jacobs, R. and Rosenbaum, P., Readings in transformational grammar*, páginas 184–221. Blaisdell, Hawai, 1970.

[11] T.T. Coelho. Resolução de anáfora pronominal utilizando o algoritmo de Lappin e Leass. Dissertação de Mestrado, Universidade Estadual de Campinas, 2005.

[12] T.T. Coelho e A.M.B.R. Carvalho. Lappin and Leass' algorithm for pronoun resolution in portuguese. Em *12th Portuguese Conference on Artificial Intelligence - EPIA 2005*, páginas 680–692. Springer, 2005.

[13] T.T. Coelho e A.M.B.R. Carvalho. Uma adaptação do algoritmo de Lappin e Leass para resolução de anáforas em português. Em *III Workshop em Tecnologia da Informação e da Linguagem Humana - TIL 2005*. Anais do Congresso Nacional da SBC 2005, São Leopoldo, RS, Julho 2005.

[14] S. Collovini, T. Carbonel, J.T. Fuchs, J.C.B. Coelho, L.H.M. Rino, e R. Vieira. Summ-it: um corpus anotado com informações discursivas visando à sumarização automática. Em *V Workshop em Tecnologia da Informação e da Linguagem Humana - TIL 2007*. Anais do Congresso Nacional da SBC 2007, Rio de Janeiro, RJ, Julho 2007.

[15] C. Cunha e L.F. Lindley. *Nova Gramática do Português Contemporâneo*. Nova Fronteira, Rio de Janeiro, RJ, 1985.

[16] C.V. Gasperin, R. Vieira, R.R.V. Goulart, e P. Quaresma. Extracting xml syntactic chunks from portuguese corpora. Em *Proceedings of the Workshop TALN 2003 Natural Language Processing of Minority Languages and Small Languages*. França, Junho 2003.

[17] B.J. Grosz, S. Weinstein, e A.K. Joshi. Centering: a framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225, 1995.

[18] M. Guedes. Parâmetro do sujeito nulo em traduções brasileiras. *Revista Rónai*, 00:36–38, 2007.

[19] M.A.K. Halliday e R. Hasan. *Coehsion in English*. Longman, Nova York, EUA, 6ª edição, 1984.

[20] G. Hirst. *Anaphora in natural language understanding: a survey*, volume 119, *Lecture notes in computer science*. Springer, 1981.

[21] J.R. Hobbs. Pronoun resolution. Relatório técnico, City University of New York, Nova York, EUA, Agosto 1976.

[22] C. Kennedy e B. Boguraev. Anaphora for everyone: pronominal anaphora resolution without a parser. Em *Proceedings of the 16th Conference on Computational Linguistics*, volume 1, páginas 113–118. Computational Linguistics, Copenhagen, Dinamarca, 1996.

[23] I.G.V. Koch. *A coesão textual*. Contexto, São Paulo, 17ª edição, 2002.

[24] S. Lappin e H.J. Leass. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–561, Dezembro 1994.

[25] V.J. Leffa. A resolução da anáfora no processamento da língua natural. Relatório técnico, Universidade Católica de Pelotas, Viamão, RS, Setembro 2001. Disponível em: <http://www.leffa.pro.br/anafor_rel.htm>. Acessado em 20 de agosto de 2007.

[26] R.M. Maduro e A.M.B.R. Carvalho. Syntactic analysis for ellipsis handling in coordinated clauses. Em *Proceedings of the 16th Brazilian Symposium on Artificial Inteligence*, páginas 397–406. Springer-Verlag, Porto de Galinhas, Recife, 2002.

[27] W.C. Mann e S.A. Thompson. Rhethorical structure theory: toward a functional theory of text organization. *Text*, 8(2), 1988.

[28] R. Mitkov. *Anaphora Resolution*. Longman, 2002.

[29] R. Mitkov. *The Oxford Handbook of Computational Linguistics*. Oxford University Press, Grã-Bretanha, 2003.

[30] C. Müller e M. Strube. Mmax: A tool for the annotation of multi-modal corpora. Em *Proceedings of the 2nd IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, páginas 45–50. Seattle, Washington, EUA, 2001.

[31] M. Palomar, A. Ferrández, L. Moreno, P. Martínez-Barco, J. Peral, M. Saiz-Noeda, e R. Muñoz. An algorithm for anaphora resolution in spanish texts. *Computational Linguistics*, 27(4):545–567, 2001.

[32] I. Paraboni e V.L.S. Lima. Possessive pronominal anaphor resolution in Portuguese written texts. Em *the 17th International Conference on Computational Linguistics*, volume 2, páginas 1010–1014. Computational Linguistics, Montreal, Quebec, Canada, 1998.

[33] M.A. Walker. Evaluating discourse processing algorithms. Em *the 27th Annual Meeting of the Association for Computational Linguistics*, páginas 251–261. Morristown, NJ, EUA, 1989.