

Universidade Federal do Paraná

Departamento de Informática

Tiago Santos de Lima

Michel Gagnon

Uma implementação da teoria de centering para resolução de
anáforas em língua portuguesa

Relatório Técnico
RT_DINF 003/2002

Curitiba, PR
2002

SUMÁRIO

1	INTRODUÇÃO	1
2	A TEORIA DE CENTERING	2
2.1	Terminologia	2
2.1.1	Transições	4
2.1.2	Restrições e regras	5
3	DESCRIÇÃO DO PROTÓTIPO	6
3.1	Reconhecimento sintático	6
3.2	Algoritmo de resolução de anáforas	9
3.3	Um exemplo de execução	10
4	LIMITAÇÕES DO PROTÓTIPO	13
4.1	Inicialização	13
4.2	Co-referentes distantes	13
4.3	Sujeito nulo	14
5	COMPARAÇÃO COM O ALGORITMO DE FOCUSING	15
5.1	Algoritmo de focusing	15
5.2	Resultados das comparações	15
6	CONCLUSÃO	17
	BIBLIOGRAFIA	18

CAPÍTULO 1

INTRODUÇÃO

É comum a utilização de anáforas em discursos da língua portuguesa. As anáforas são referências indiretas a elementos presentes no contexto como, por exemplo, pronomes pessoais. É fácil observar que não é possível sua utilização indiscriminada, sob pena de tornar o discurso ambíguo ou até mesmo incompreensível. Desta forma, percebe-se que o emprego de anáforas está limitado na coerência do discurso.

Neste trabalho apresento uma implementação do algoritmo de Brennan, Friedman e Pollard descrito em [1]. Este algoritmo resolve anáforas utilizando a teoria de *centering* como base. Descrita por Grosz, Joshi e Weinstein em [4], a teoria de centering apresenta regras que regem a interconexão entre enunciados de um discurso. Seguindo estas regras, o algoritmo consegue resolver as anáforas presentes.

Foi feita também uma pequena comparação com uma implementação do algoritmo de *focusing* outrora desenvolvido e apresentado em [6]. O algoritmo de focusing procura antecipar quais as entidades do discurso a serem indiretamente referenciadas nos enunciados subsequentes. Esta abordagem, embora similar a de centering, traz alguns resultados diferentes.

O restante deste trabalho está organizado da seguinte forma: O primeiro capítulo apresenta a teoria de centering, sua terminologia e demais definições. Em seguida, o algoritmo de Brennan, Friedman e Pollard que foi implementado é ilustrado com um exemplo de sua execução. O protótipo implementado apresenta algumas limitações que são descritos no capítulo 4, juntamente com algumas sugestões para contornar os mesmos, quando possível. A comparação entre os algoritmos de centering e focusing implementados é feita no capítulo 5. Uma breve explanação do algoritmo de focusing também é dada neste capítulo.

CAPÍTULO 2

A TEORIA DE CENTERING

De acordo com a teoria de *centering*, um discurso deve apresentar coerência na seqüência de enunciados que o forma. Um discurso deve exibir coerência global – entre os seus diversos segmentos – e coerência local – entre as declarações de um mesmo segmento. A teoria de centering propõe um modelo para o tratamento da coerência local descrevendo um sistema de restrições e regras que governam as relações entre o foco de atenção do discurso e as formas escolhidas para construção das declarações que o compõem. Ilustrei os conceitos descritos através de dois exemplos adaptados de Grosz, Joshi e Weinstein ([4]):

- (1)
 - (a) João foi a sua loja de música favorita para comprar um piano.
 - (b) Ele havia freqüentado a loja por vários anos.
 - (c) Estava excitado porque iria finalmente poder comprar um piano.
 - (d) Mas quando chegou, a loja estava fechada.

- (2)
 - (a) João foi a sua loja de música favorita para comprar um piano.
 - (b) Esta era a loja que João freqüentou por vários anos.
 - (c) Ele estava excitado porque iria finalmente poder comprar um piano.
 - (d) Ela estava fechada quando João chegou.

Os dois segmentos de discurso expressam exatamente a mesma informação utilizando enunciados diferentes. No entanto, o discurso (1) é intuitivamente mais coerente que o discurso (2). Isso parece acontecer porque no primeiro caso trata-se apenas de um indivíduo central, João, enquanto que no segundo, o foco principal oscila entre João e a loja de música. Isto mostra que diferentes formas sintáticas implicam em diferenças na inferência dos referentes anafóricos para o leitor (ou ouvinte). A teoria de centering fornece as bases para tratamento destas diferenças. Isto foi usado por Brennan, Friedman e Pollard, em [1], para desenvolvimento de um algoritmo capaz de resolver anáforas.

2.1 Terminologia

Um segmento de discurso consiste de uma seqüência de enunciados U_1, U_2, \dots, U_m . Os enunciados possuem a propriedade de *realizar* entidades do contexto do discurso. Por exemplo, no enunciado João foi a sua loja de música favorita comprar um piano., temos como entidades realizadas, JOAO, LOJA-DE-MUSICA e PIANO.

Os *centros* representam as entidades do mundo referidas pela sentença atual. Estes objetos servem de ligação entre um enunciado e outro no segmento de discurso que os contém. Centros são objetos semânticos, não são palavras, frases ou formas sintáticas. Além disso, o mesmo enunciado pode possuir centros diferentes em situações diferentes.

Para cada enunciado U_n é associada um conjunto ordenado de *centros prospectivos* (*Forward-looking centers*), $C_f(U_n)$, consistindo das entidades do discurso que são realizadas por este enunciado.

A ordem dos elementos de $C_f(U_n)$ segue o seguinte critério: para todo $f_i, f_j \in C_f(U_n)$, se f_i realiza um sujeito e f_j realiza um objeto, então $f_i \prec f_j$. Ou seja, f_i precede f_j . O primeiro elemento de $C_f(U_n)$ é chamado *próximo centro preferencial* (*Preferred center*), $C_p(U_n)$.

O enunciado U_n , ou enunciado atual *está centrado*, isto é, discorre sobre uma única entidade chamada de *centro retrospectivo* (*Backward-looking center*), $C_b(U_n)$, que é realizado pelo enunciado imediatamente anterior, U_{n-1} .

Abaixo, um exemplo que serve como ilustração dos termos introduzidos:

- (3) (a) João_{a₁} possui um BMW_{a₂}.
 (b) Ele_{r₁} dirige rápido.
 (c) Pedro_{a₃} corre com ele_{r₂} no feriado_{a₄}.
 (d) Ele_{r₃} freqüentemente o_{r₄} vence.

Como pode ser observado no exemplo, as entidades do mundo são reconhecidas através dos substantivos que as descrevem: João, BMW, Pedro e feriado. Os índices que aparecem nos enunciados (a_1 , a_2 , a_3 e a_4) nomeiam as construções do segmento de discurso que referenciam as entidades. Da mesma forma, nomeia-se os elementos anafóricos encontrados (r_1 , r_2 , r_3 e r_4). Com a classificação completada, pode-se construir a *âncora*, o par $\langle C_b, C_f \rangle$, de cada enunciado:

- (3') (a) $\langle (?), [(JOAO, a_1), (BWM, a_2)] \rangle$
 (b) $\langle (JOAO, a_1), [(JOAO, r_1)] \rangle$
 (c) $\langle (JOAO, r_1), [(PEDRO, a_3), (JOAO, r_2), (FERIADO, a_4)] \rangle$
 (d) $\langle (PEDRO, a_3), [(PEDRO, r_3), (JOAO, r_4)] \rangle$

O C_b apenas indica quem é a atual entidade central do discurso. No conjunto C_f por sua vez, temos quais entidades foram realizadas por quais elementos. O par (PEDRO, r_3) da âncora (d), por exemplo, indica que a entidade PEDRO foi realizada pelo elemento r_3 . A primeira âncora apresenta um (?) como C_b . Isto acontece porque a teoria não define como escolher o C_b do primeiro enunciado do segmento de discurso.

2.1.1 Transições

Outro conceito importante da teoria são as transições entre os enunciados. Elas descrevem como as estes são ligados em um segmento de discurso coerente. Grosz, Joshi e Weinstein ([4]) propuseram três tipos de transições. No entanto, o algoritmo construído posteriormente por Brennan Friedman e Pollard ([1]) propõe quatro tipos, são eles:

1. **Continue:** $C_b(U_{n-1}) = C_b(U_n) = C_p(U_n)$.

O discurso permanece centrado na mesma entidade e esta é o centro preferido a ser usado na próxima sentença.

2. **Retain:** $C_b(U_{n-1}) = C_b(U_n) \neq C_p(U_n)$.

O discurso permanece centrado na mesma entidade, mas na próxima sentença um novo centro será o preferido.

3. **Smooth-shift:** $C_b(U_{n-1}) \neq C_b(U_n) = C_p(U_n)$.

O discurso trocou de centro e este novo centro é o preferido para ser usado na próxima sentença.

4. **Rough-shift:** $C_b(U_{n-1}) \neq C_b(U_n) \neq C_p(U_n)$.

O discurso trocou de centro e um outro centro será o preferido a ser usado na próxima sentença.

Abaixo, um exemplo com os tipos de transições descritos:

- (4) (a) Pedro_{a₁} é um cantor.
 $\langle (?), [(PEDRO, a_1)] \rangle$
- (b) Ele_{r₁} visitou Maria_{a₂} ontem.
 $\langle (PEDRO, a_1), [(PEDRO, r_1), (MARIA, a_2)] \rangle$
- (ci) Ele_{r₂} conversou muito com ela_{r₃}
Continue: $\langle (PEDRO, r_1), [(PEDRO, r_2), (MARIA, r_3)] \rangle$
- (cii) Ela_{r₂} recebeu a visita dele_{r₃} entusiasmada.
Retain: $\langle (PEDRO, r_1), [(MARIA, r_2), (PEDRO, r_3)] \rangle$
- (ciii) Ela_{r₂} não gostou.
Smooth-shift: $\langle (MARIA, a_2), [(MARIA, r_2)] \rangle$
- (civ) Joana_{a₃} a_{r₂} viu na semana passada.
Rough-shift: $\langle (MARIA, a_2), [(JOANA, a_3), (MARIA, r_2)] \rangle$

2.1.2 Restrições e regras

O sistema proposto também contém as seguintes restrições e regras:

- **Restrições**

1. Existe apenas um C_b .
2. Todo elemento de $C_f(U_n)$ deve ser realizado em U_n .
3. $C_b(U_n)$ é o mais bem colocado elemento de $C_f(U_{n-1})$ que é realizado em U_n .

- **Regras**

1. Se $f_j \in C_f(U_{n-1})$ e f_j é realizado por um pronome em U_n , então todo $f_i \in C_f(U_{n-1})$ realizado em U_n tal que $f_i \prec f_j$ em $C_f(U_{n-1})$, deve ser realizado por um pronome em U_n . Isto implica que se existe um pronome na sentença, então C_b é realizado por um pronome.
2. *Continue* tem preferência sobre *retain* que tem preferência sobre *smooth-shift* o qual tem preferência sobre *rough-shift*.

Além das regras e restrições apresentadas acima, o algoritmo também utiliza um filtro chamado de *contra-índice*. Ele se baseia no fato de que vários verbos não podem ser reflexivos sem presença de um pronome reflexivo. Assim como, se o seu objeto for um pronome reflexivo, este deverá ser realizado com o agente. Abaixo, dois exemplos:

(5) José_i estava jogando futebol quando Pedro_j o_i viu no campo.

(6) José_i estava jogando futebol quando Pedro_j se_j viu no espelho.

Nos enunciados acima, temos que o objeto do verbo *ver* não pode ser o próprio agente se este não for um pronome reflexivo. Ao aplicarmos o filtro de contra-índice em (5) *Pedro* não é mais considerado um antecedente possível. Em (6), no entanto, o antecedente precisa ser o agente. Neste caso *José* é desconsiderado ao aplicarmos o filtro.

CAPÍTULO 3

DESCRIÇÃO DO PROTÓTIPO

Neste capítulo descrevo o protótipo implementado para resolução de anáforas. Este protótipo faz a resolução das anáforas em discursos em dois passos: primeiramente, utiliza uma gramática para reconhecimento sintático e em seguida, o algoritmo de centering para resolver as anáforas.

3.1 Reconhecimento sintático

A análise sintática do discurso é feita com uma *Gramática de Unificação*, como descrito por Gazdar, Ewan, Pullum e Sag em [3]. Os traços lexicais das palavras da linguagem são utilizados num processo de verificação de consistência em cada nó da árvore sintática. Eles asseguram que as sentenças sejam reconhecidas somente se a concordância estiver correta. Além disso, são responsáveis por passar as informações necessárias para o algoritmo de resolução de anáforas. Abaixo, um fragmento da gramática e do dicionário:

Gramática:

$$\begin{array}{l}
 \text{(G1)} \quad \begin{array}{ccc} D & \rightarrow & S \\ \left[\begin{array}{l} utts : [1] \\ con : [3] \end{array} \right] & & \left[\begin{array}{l} utt : [1] \\ con : [3] \end{array} \right] \end{array} \\
 \text{(G2)} \quad \begin{array}{ccccc} D & \rightarrow & S & C & D \\ \left[\begin{array}{l} utts : [1] + [3] \\ con : [2] + [4] \end{array} \right] & & \left[\begin{array}{l} utt : [1] \\ con : [2] \end{array} \right] & & \left[\begin{array}{l} utts : [3] \\ con : [4] \end{array} \right] \end{array} \\
 \text{(G3)} \quad \begin{array}{ccccc} S & \rightarrow & SN & & SV \\ \left[\begin{array}{l} utt : ([2] + [5], [3] + [6]) \\ con : [4] + [7] \end{array} \right] & & \left[\begin{array}{l} num : [1] \\ cen : \left[\begin{array}{l} ref : [2] \\ ant : [3] \\ con : [4] \end{array} \right] \end{array} \right] & & \left[\begin{array}{l} num : [1] \\ cen : \left[\begin{array}{l} ref : [5] \\ ant : [6] \\ con : [7] \end{array} \right] \end{array} \right] \end{array} \\
 \text{(G4)} \quad \begin{array}{ccc} SN & \rightarrow & N \\ \left[\begin{array}{l} num : [1] \\ cen : [2] \end{array} \right] & & \left[\begin{array}{l} num : [1] \\ cen : [2] \end{array} \right] \end{array} \\
 \text{(G6)} \quad \begin{array}{ccccc} SV & \rightarrow & V & & SN \\ \left[\begin{array}{l} num : [1] \\ cen : [2] \end{array} \right] & & \left[\begin{array}{l} num : [1] \end{array} \right] & & \left[\begin{array}{l} num : [1] \\ cen : [2] \end{array} \right] \end{array}
 \end{array}$$

Dicionário:

- (D1) $\langle \text{João}, N \rangle$
 $\left[\begin{array}{l} \text{num} : \text{sin} \\ \text{cen} : \left[\begin{array}{l} \text{ref} : [] \\ \text{ant} : \text{JOAO} \\ \text{con} : [\text{HOMEM}(\text{JOAO})] \end{array} \right] \end{array} \right]$
- (D2) $\langle \text{Maria}, N \rangle$
 $\left[\begin{array}{l} \text{num} : \text{sin} \\ \text{cen} : \left[\begin{array}{l} \text{ref} : [] \\ \text{ant} : \text{MARIA} \\ \text{con} : [\text{MULHER}(\text{MARIA})] \end{array} \right] \end{array} \right]$
- (D3) $\langle \text{ela}, N \rangle$
 $\left[\begin{array}{l} \text{num} : \text{sing} \\ \text{cen} : \left[\begin{array}{l} \text{ref} : x \\ \text{ant} : [] \\ \text{con} : [\text{FEMININO}(x)] \end{array} \right] \end{array} \right]$
- (D3) $\langle \text{o}, N \rangle$
 $\left[\begin{array}{l} \text{num} : \text{sing} \\ \text{cen} : \left[\begin{array}{l} \text{ref} : y \\ \text{ant} : [] \\ \text{con} : [\text{MASCULINO}(y)] \end{array} \right] \end{array} \right]$
- (D4) $\langle \text{ama}, V \rangle$
 $\left[\text{num} : \text{sin} \right]$
- (D5) $\langle e, C \rangle$

Esta gramática é formada por regras de produção cujos elementos possuem um conjunto de propriedades chamadas *estruturas de traços*. As regras gramaticais são aplicadas de acordo com um sistema de regras de produção comum. Neste caso porém, para que uma regra seja aplicada, as variáveis, simbolizadas com um número dentro de um pequeno quadrado, devem ser *unificadas* corretamente. Ou seja, caso a variável esteja *livre*, ainda não foi iniciada com nenhum valor, ela recebe o valor de sua correspondente (a que possui o mesmo número), caso contrário as duas variáveis devem possuir o mesmo valor. Há dois tipos de traços: os traços usados para verificação sintática, como o traço *num*, e o traço *cen*, usado para passar as informações ao algoritmo de resolução de anáforas. Ilustro o funcionamento desta gramática através de um pequeno exemplo:

- (7) (a) João ama Maria
 (b) e ela o ama.

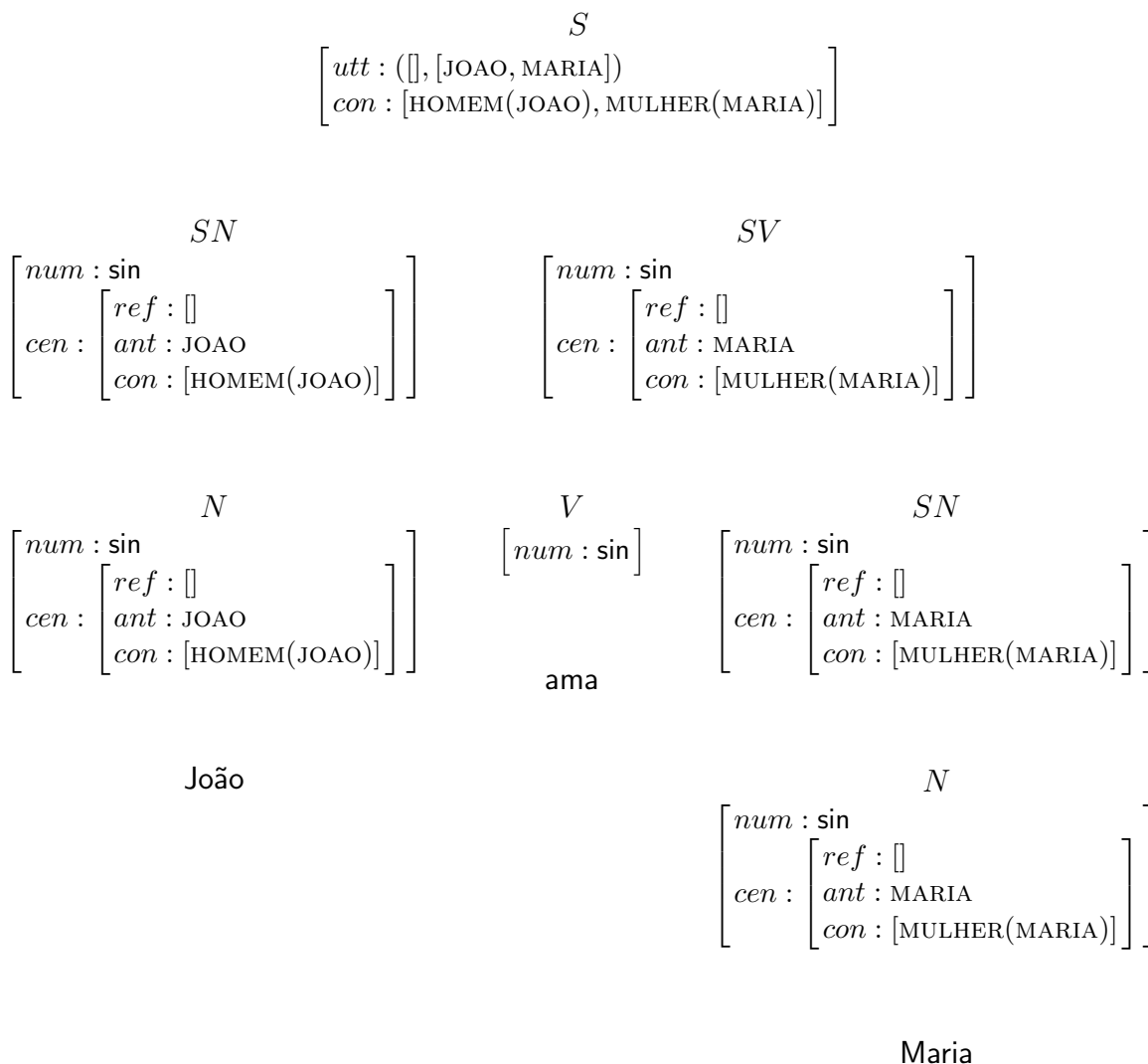


Figura 3.1: Árvore sintática da sentença João ama Maria.

Primeiramente é feito um mapeamento dos itens léxicos a termos do dicionário. Na figura 3.1 pode-se ver este passo sendo feito nas folhas da árvore sintática: as variáveis são instanciadas com o valor dos traços definidos para este item. As unificações vão sendo feitas de baixo para cima na árvore, que é construída seguindo-se as regras de produção da gramática. No momento da aplicação da regra (G3), os valores dos traços *num* são unificados corretamente. Observe também que as listas *ref* *ant* e *con* são construídas no decorrer da análise sintática.

Na figura 3.2, podemos ver a árvore sintática do restante do discurso. A segunda sentença é analisada analogamente à primeira e as regras (G2) e (G1) são aplicadas. No nodo *S*, as listas *ref* e *ant* formam uma tupla chamada *utt*. O discurso por sua vez, possui uma lista de tuplas chamada *utts* e uma lista de restrições *con*. Estas listas serão utilizadas mais adiante pelo algoritmo de resolução de anáforas.

- (b) Construa a lista de possíveis $C_f(U_n)$ como sendo $P_{C_f} = A_{r_1} \times A_{r_2} \times \dots \times A_{r_m}$, onde $A_{r_j} = A \times [r_j]$ para $r_j \in R$ (mantendo a ordem dos elementos).
- (c) Filtre P_{C_f} utilizando o conhecimento do mundo (lista *con*).
- (d) Construa a lista de possíveis $C_b(U_n)$ como sendo $P_{C_b} = C_f(U_{n-1}) + [\text{NIL}]$.
- (e) Finalmente, construa o conjunto de possíveis âncoras $\mathcal{A} = P_{C_f} \times P_{C_b}$.

2. Filtre o conjunto \mathcal{A} :

- (a) Elimine todos os pares que não respeitam o filtro de contra-índice.
- (b) Elimine todos os pares que não respeitam a restrição 3.
- (c) Elimine todos os pares que não respeitam a regra 1.

3. Classifique e ordene as âncoras utilizando a regra 2.

4. Escolha a âncora mais bem colocada.

3.3 Um exemplo de execução

Vou ilustrar o funcionamento do algoritmo proposto utilizando um exemplo adaptado de [1]:

Segmento de discurso: Primeiramente, vou apresentar o segmento de discurso que antecede o enunciado atual, bem como sua análise realizada previamente:

- (a) João_{a₁} possui um BMW_{a₂}.
- (b) Ele_{r₁} dirige rápido.
- (c) Pedro_{a₃} corre com ele_{r₂} no feriado_{a₄}.

Âncoras: (análise prévia)

- (a) $\langle\langle \text{JOAO}, a_1 \rangle, [(\text{JOAO}, a_1), (\text{BWM}, a_2)] \rangle$
- (b) $\langle\langle \text{JOAO}, a_1 \rangle, [(\text{JOAO}, r_1)] \rangle$
- (c) $\langle\langle \text{JOAO}, r_1 \rangle, [(\text{PEDRO}, a_2), (\text{JOAO}, r_2), (\text{FERIADO}, a_4)] \rangle$

Declaração atual (U_n): Enunciado que será aplicado ao algoritmo descrito:

Ele_{r₃} freqüentemente o_{r₄} vence.

Execução do Algoritmo:

1. Seja $U_n = ([r_3, r_4], [])$, construa o conjunto \mathcal{A} :

(a) $A = A_{U_{n-1}} + A_{U_n} = [\text{PEDRO}, \text{JOAO}, \text{FERIADO}]$

(b) $P_{C_f} =$

$$[(\text{PEDRO}, r_3), (\text{PEDRO}, r_4)]$$

$$[(\text{PEDRO}, r_3), (\text{JOAO}, r_4)]$$

$$[(\text{PEDRO}, r_3), (\text{FERIADO}, r_4)]$$

$$[(\text{JOAO}, r_3), (\text{PEDRO}, r_4)]$$

$$[(\text{JOAO}, r_3), (\text{JOAO}, r_4)]$$

$$[(\text{JOAO}, r_3), (\text{FERIADO}, r_4)]$$

$$[(\text{FERIADO}, r_3), (\text{PEDRO}, r_4)]$$

$$[(\text{FERIADO}, r_3), (\text{JOAO}, r_4)]$$

$$[(\text{FERIADO}, r_3), (\text{FERIADO}, r_4)]$$

(c) Todas as âncoras que não representam situações possíveis de acordo com o conhecimento do mundo introduzido são eliminados. Por exemplo, o algoritmo rejeita entidades não humanas realizadas por pronomes pessoais. Logo, o conjunto filtrado fica:

$$[(\text{PEDRO}, r_3), (\text{PEDRO}, r_4)]$$

$$[(\text{PEDRO}, r_3), (\text{JOAO}, r_4)]$$

$$[(\text{JOAO}, r_3), (\text{PEDRO}, r_4)]$$

$$[(\text{JOAO}, r_3), (\text{JOAO}, r_4)]$$

(d) $P_{C_b} = [(\text{PEDRO}, a_2), (\text{JOAO}, r_2), (\text{FERIADO}, a_4), (\text{NIL}, \text{NIL})]$

(e) Conjunto \mathcal{A} :

i. $\langle (\text{PEDRO}, a_2), [(\text{PEDRO}, r_3), (\text{PEDRO}, r_4)] \rangle$

ii. $\langle (\text{PEDRO}, a_2), [(\text{PEDRO}, r_3), (\text{JOAO}, r_4)] \rangle$

iii. $\langle (\text{PEDRO}, a_2), [(\text{JOAO}, r_3), (\text{JOAO}, r_4)] \rangle$

iv. $\langle (\text{PEDRO}, a_2), [(\text{JOAO}, r_3), (\text{PEDRO}, r_4)] \rangle$

v. $\langle (\text{JOAO}, r_2), [(\text{PEDRO}, r_3), (\text{PEDRO}, r_4)] \rangle$

vi. $\langle (\text{JOAO}, r_2), [(\text{PEDRO}, r_3), (\text{JOAO}, r_4)] \rangle$

vii. $\langle (\text{JOAO}, r_2), [(\text{JOAO}, r_3), (\text{JOAO}, r_4)] \rangle$

viii. $\langle (\text{JOAO}, r_2), [(\text{JOAO}, r_3), (\text{PEDRO}, r_4)] \rangle$

ix. $\langle (\text{FERIADO}, a_4), [(\text{PEDRO}, r_3), (\text{PEDRO}, r_4)] \rangle$

x. $\langle (\text{FERIADO}, a_4), [(\text{PEDRO}, r_3), (\text{JOAO}, r_4)] \rangle$

xi. $\langle (\text{FERIADO}, a_4), [(\text{JOAO}, r_3), (\text{JOAO}, r_4)] \rangle$

xii. $\langle (\text{FERIADO}, a_4), [(\text{JOAO}, r_3), (\text{PEDRO}, r_4)] \rangle$

xiii. $\langle (\text{NIL}, \text{NIL}), [(\text{PEDRO}, r_3), (\text{PEDRO}, r_4)] \rangle$

xiv. $\langle (\text{NIL}, \text{NIL}), [(\text{PEDRO}, r_3), (\text{JOAO}, r_4)] \rangle$

- xv. $\langle (\text{NIL}, \text{NIL}), [(\text{JOAO}, r_3), (\text{JOAO}, r_4)] \rangle$
 xvi. $\langle (\text{NIL}, \text{NIL}), [(\text{JOAO}, r_3), (\text{PEDRO}, r_4)] \rangle$

2. Filtre o conjunto \mathcal{A} :

- (a) remove todas as âncoras com $C_f = [(\text{PEDRO}, r_3), (\text{PEDRO}, r_4)]$: i, v, ix e xiii
 remove também as âncoras com $C_f = [(\text{JOAO}, r_3), (\text{JOAO}, r_4)]$: iii, vii, xi e xv
- (b) remove as âncoras com $C_b = (\text{JOAO}, r_2)$: v, vi, vii e viii
 remove também as âncoras com $C_b = (\text{FERIADO}, a_4)$: ix, x, xi e xii
 remove também as âncoras com $C_b = (\text{NIL}, \text{NIL})$: xiii, xiv, xv, xvi
- (c) Sem efeito, pois nenhum par desrespeita a regra 1.

Conjunto de âncoras resultante:

- ii. $\langle (\text{PEDRO}, a_2), [(\text{PEDRO}, r_3), (\text{JOAO}, r_4)] \rangle$
 iv. $\langle (\text{PEDRO}, a_2), [(\text{JOAO}, r_3), (\text{PEDRO}, r_4)] \rangle$

3. Classifique e ordene as âncoras:

- ii = smooth-shift
 iv = rough-shift

4. Vamos escolher a âncora ii e, portanto, temos como resultado:

$$C_b = (\text{PEDRO}, a_2) \text{ e } C_f = [(\text{PEDRO}, r_3), (\text{JOAO}, r_4)]$$

$$\text{Ele}_{r_3} = \text{PEDRO}$$

$$\text{o}_{r_4} = \text{JOAO}$$

CAPÍTULO 4

LIMITAÇÕES DO PROTÓTIPO

Nem todos os fenômenos anafóricos possíveis podem ser tratados pelo algoritmo implementado. Nesta seção apresento algumas das limitações mais importantes.

4.1 Inicialização

A teoria de centering e também a descrição do algoritmo de [1] não deixam claro como deve ser tratada o primeiro enunciado do segmento do discurso.

Uma saída possível para este problema seria a utilização do *foco global* do discurso. Este foco é aquele que se estabelece no início do discurso e permanece entre os seus segmentos. As entidades presentes no foco global poderiam também ser concatenadas nas listas C_f de cada enunciado. O efeito colateral seria a necessidade de definirmos novos tipos de transições entre os enunciados do segmento. Um problema similar ocorre com os *dêiticos* “eu”, “tu”, “nós” e “você”. Neste caso, as informações de discursante e ouvinte poderiam ser concatenadas nas listas C_f .

4.2 Co-referentes distantes

O algoritmo descrito não consegue resolver o antecedente das anáforas onde os co-referentes estejam distantes. Isso ocorre quando o foco de atenção é deixado de lado por um momento. O foco de atenção pode ser “abandonado” para apresentar alguma explicação ou argumentação. Como se abrissemos um parênteses e, em seguida, retomássemos o discorrer normal das idéias. Coloco aqui um exemplo para ilustrar estes casos:

- (8) (a) Pedro_{*i*} comprou um relógio_{*j*} novo.
 (b) Este relógio_{*j*} é bem mais bonito que o antigo.
 (c) Ele_{*i*} o_{*j*} adora.

O pronome Ele deve ser resolvido com Pedro, o qual não aparece no enunciado (b). Como a escolha do $C_b(U_n)$ é restrita aos centros realizados pelo último enunciado, o algoritmo não é capaz de resolver esta anáfora.

4.3 Sujeito nulo

Os exemplos colocados a seguir, apresentados por Petry em [7] (veja também [2]), mostram que a substituição de um pronome por uma elipse pode afetar a interpretação de um discurso:

(9) Quando João_{*i*} encontrou Pedro_{*j*}, ele_{*j*} nem ao menos lhe_{*k*} disse oi.

(10) Quando João_{*i*} encontrou Pedro_{*j*}, \emptyset _{*i*} nem ao menos lhe_{*j*} disse oi.

No primeiro caso, o pronome *ele* explicitamente colocado, provoca a interpretação de que foi João quem não cumprimentou Pedro, enquanto que no segundo caso ocorre o inverso. Esta sutileza não é detectada pelo algoritmo implementado.

CAPÍTULO 5

COMPARAÇÃO COM O ALGORITMO DE FOCUSING

A comparação que faço nesta seção se dá com um algoritmo implementado em outra oportunidade que utiliza a teoria de *focusing* criada por Sidner em [10] (veja também [5, 8, 9]). Este último, completamente descrito em [6], faz uso de estruturas DRSs para compor a representação semântica de discursos com anáforas.

5.1 Algoritmo de focusing

O algoritmo de focusing trabalha na tentativa de prever quais são as entidades mais prováveis a serem referidas nas sentenças subsequentes. No início do discurso, a primeira sentença é utilizada para iniciar o *foco de discurso* e o *foco ator*, duas pilhas mantidas em memória. O foco de discurso é aquele preferível a ser referenciado em posição de objeto das sentenças subsequentes. O ator é aquele que provavelmente será referenciado em posição de agente. Ao final da análise de cada sentença subsequente, são aplicadas regras para atualização do foco:

1. Foco ator:

- (a) se a sentença possui um novo agente, adicione este agente no compartimento de focos atores.
- (a) caso contrário nada precisa ser feito.

2. Foco de discurso:

- (a) se já existe um foco de discurso e uma anáfora ligando-o, então o foco fica como estava.
- (b) caso contrário, se a sentença possui um objeto adicione-o no compartimento de focos de discurso.
- (a) caso contrário, nada precisa ser feito.

5.2 Resultados das comparações

A primeira diferença que notada, é que o algoritmo de focusing não apresenta o problema da inicialização descrito na seção 4.1. Deste modo, discursos contendo anáforas já na primeira sentença podem ser resolvidos por este algoritmo, excluindo-se aqueles que fazem uso de dêiticos ou referem-se a elementos ainda não introduzidos no discurso.

Outro problema não existente no algoritmo de focusing é o descrito na seção 4.2. Este algoritmo tem a vantagem de não observar apenas a sentença imediatamente anterior para detectar os elementos centrais do discurso.

Há porém, um exemplo em que o algoritmo de focusing não encontra o antecedente correto para uma referência anafórica, e que centering o faz:

- (11) (a) Pedro_i cozinhou arroz_j e
 (b) o_i descartou \emptyset _j.
 (c) \emptyset _i Cozinhou feijão_k,
 (d) \emptyset _i esperou e o_k comeu.

Focusing inicializa o foco de discurso com **arroz** na primeira sentença. Em seguida, na segunda sentença, temos uma anáfora ligando-o. Por esta razão, a regra 2 é utilizada na atualização do foco de discurso entre as sentenças (c) e (d). Logo em seguida, o algoritmo liga o pronome **o** da última sentença com **arroz** em vez de **feijão**. Tal problema não ocorre com centering porque o substantivo **arroz** nem sequer aparece na sentença (c). O substantivo **arroz** não faz parte da lista $C_f(U_{n-1})$, o que torna impossível ligá-lo com **o**. A ligação é feita corretamente com o centro atual, **feijão**.

CAPÍTULO 6

CONCLUSÃO

Por ser um fenômeno freqüente em várias linguagens naturais, as anáforas recebem tratamento diverso em teorias distintas. Este trabalho descreve a implementação de um algoritmo para tratamento destes fenômenos no Português do Brasil. A teoria escolhida, centering, é foco de pesquisa de muitos trabalhos na área de resolução de anáforas atualmente, dada a quantidade de publicações recentes a seu respeito. Isso demonstra o potencial de centering para se tornar uma teoria mais completa neste sentido.

Embora a teoria de centering pareça ser mais “elegante”, a característica combinatorial dos cálculos empregados descrevem um algoritmo muito menos eficiente do que focusing, ao menos do ponto de vista de recursos computacionais. Centering também demonstrou possuir alguns pequenos problemas contornáveis, pelo menos aparentemente. A inicialização dos centers e co-referentes distantes são bons exemplos.

A comparação feita com o algoritmo de focusing pretendia apontar diferenças na resolução dos pronomes das sentenças, na medida que a escolha do elemento mais saliente do discurso se faz de maneira diferenciada. Para que esta comparação fosse justa, utilizei exatamente a mesma gramática, o mesmo conhecimento de mundo e as mesmas restrições ontológicas nos dois protótipos.

A comparação porém, não salienta uma grande diferença entre as duas teorias. Como o algoritmo de focusing já conseguia uma quantidade de acertos muito boa no trabalho de Lima ([6]), podemos dizer que, até onde pude verificar, a implementação utilizando centering não perde em muito para a primeira. Vale lembrar que o conjunto de testes permitidos pelo dicionário e gramáticas implementados não é muito grande. Além disso, ele não abrange uma parcela considerável de casos da linguagem. Por esta razão, uma sugestão para trabalho futuro surge neste ponto: Analisar qual seria o comportamento destes dois algoritmos se pudermos aplicá-los a um dicionário e gramática mais completos.

BIBLIOGRAFIA

- [1] Susan E. Brennan, Walker Friedmanand, e Carl J. Pollard. A centering approach to pronouns. *25th Annual Meeting of the Association of Computational Linguistics*, páginas 155–162, 1987.
- [2] Barbara Di Eugenio. Centering theory and the italian pronominal system. *COLLING90*, 1990.
- [3] Gerald Gazdar, Klein Ewan, Geoffrey K. Pullum, e Ivan Sag. *Generalized Phrase Structure Grammar*. Atlas, Oxford, 1985.
- [4] Barbara J. Grosz, Aravind K. Joshi, e Scott Weinstein. Centering: A framework for modelling the local coherence of discourse. *Computational Linguistics*, volume 21, páginas 203–225, junho de 1995.
- [5] Barbara J. Grosz e Candace Sidner. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3), 1988.
- [6] Tiago Santos de Lima. Abordagem composicional para interpretação semântica de discurso em língua portuguesa. Trabalho de graduação, 2001.
- [7] Tatiana de Oliveira Petry. Uso da teoria de centering no tratamento computacional de referências anafóricas pronominais no Português escrito: Uma experimentação com pronomes pessoais. Dissertação de Mestrado, PUC-RS, Porto Alegre, 2000.
- [8] Candace L. Sidner. Focusing for interpretation of pronouns. *American Journal of Computational Linguistics*, 7(4):217–231, 1981.
- [9] Candace L. Sidner. Focusing in the comprehension of definite anaphora. Robert C. Berwick e Michael Brady, editors, *Computational Models of Discourse*, capítulo 5, páginas 267–330. MIT Press, Cambridge, MA, 1983. Q335 .C56 1983.
- [10] Candance L. Sidner. Towards a computational theory of definite anaphora comprehension in English discourse. Relatório Técnico AI-TR-537, MIT, 1979.