



## SEMÂNTICA – PARTE 2

*SCC5908 Tópicos em Processamento de Língua Natural*

Thiago A. S. Pardo

### ANÁLISE SEMÂNTICA

- Expressão linguística → representação do significado
  - São necessários para automação do processo
    - Grande variedade de fontes de conhecimento
      - Significado de palavras, significado associado a construções gramaticais, conhecimento da estrutura discursiva, conhecimento de senso comum, conhecimento do estado de coisas no mundo
    - Técnicas de inferência

## ANÁLISE SEMÂNTICA

- Abordagem comum: **análise semântica dirigida pela sintaxe** (*syntax-driven semantic analysis*)
  - Conhecimento proveniente do **léxico** e da **gramática**, sentencial
    - Em um primeiro momento, sem uso de informação contextual e de inferências
      - Significado literal

3

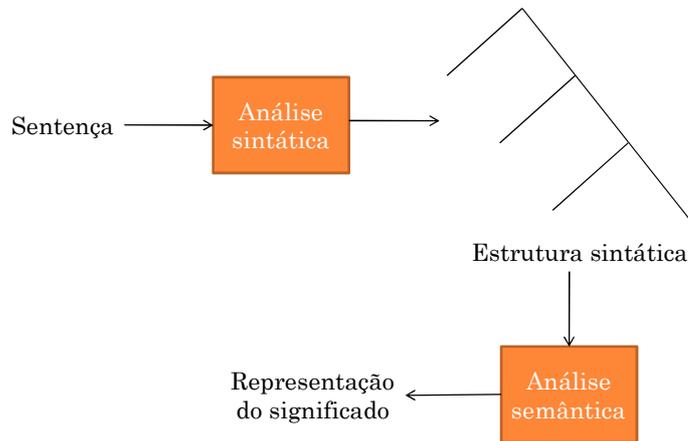
## ANÁLISE DIRIGIDA PELA SINTAXE

- **Princípio da composicionalidade**
  - **Significado da sentença** derivado do **significado das palavras**
    - Nem sempre verdade
  - **Ordem** e **agrupamento** de palavras na sentença
  - **Relações** entre palavras na sentença
    - Portanto, o significado de uma sentença é fortemente influenciado pela sintaxe

4

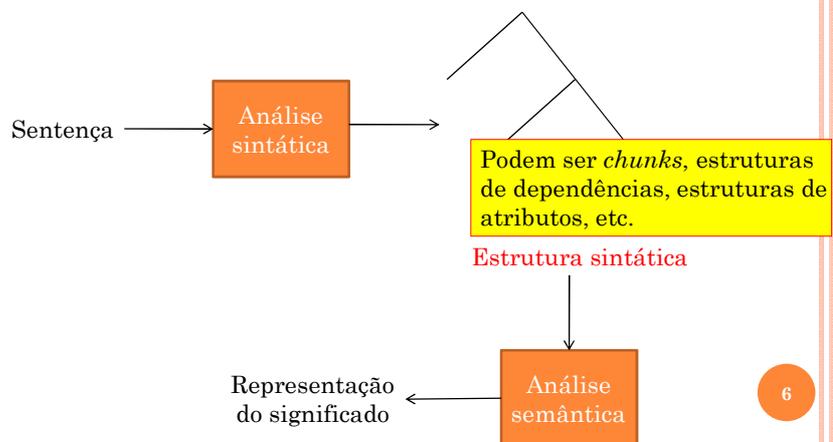
## ANÁLISE DIRIGIDA PELA SINTAXE

- Esquema simples em *pipeline*



## ANÁLISE DIRIGIDA PELA SINTAXE

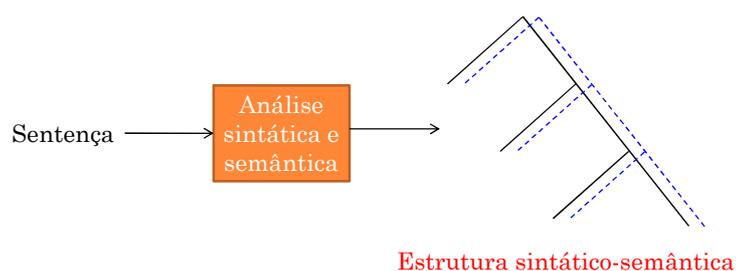
- Esquema simples em *pipeline*



## ANÁLISE DIRIGIDA PELA SINTAXE

### ○ Análise integrada

- Pode ajudar a restringir as possibilidades sintáticas
- Mas pode haver muitas inferências/análises semânticas intermediárias desnecessárias



7

## ANÁLISE DIRIGIDA PELA SINTAXE

### ○ **Ambigüidade** pode atrapalhar

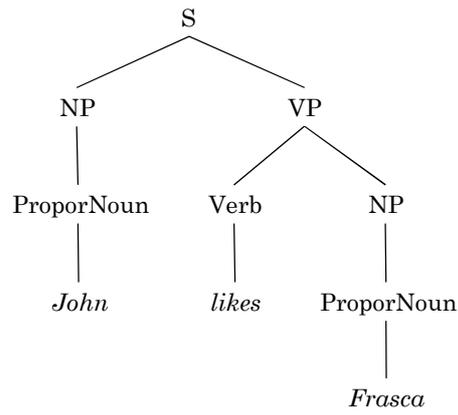
- Léxica, sintática, discursiva (correferências, por exemplo)

### ○ Em princípio, não será um problema

- Analizam-se todas as possibilidades
- Cada interpretação pode ser analisada e produzida independentemente das demais

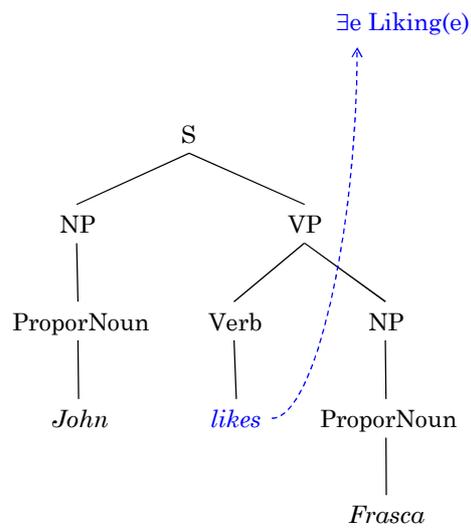
8

EXEMPLO



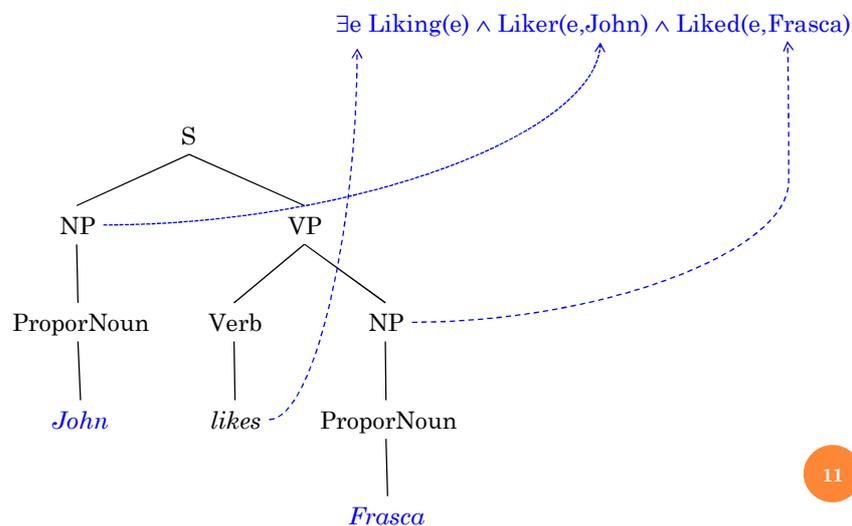
9

EXEMPLO



10

## EXEMPLO



11

## PASSO A PASSO

## ○ Análise

1. Monta-se o **esqueleto** da representação com base no **verbo**
  - Precisam-se conhecer os possíveis *templates* dos verbos
2. Recuperam-se as **representações** correspondentes aos **sintagmas nominais/complementos/argumentos** do verbo
  - Precisa-se saber onde estão e quais são os complementos/argumentos do verbo, e quais ocupam que papéis no template do verbo
3. Instanciação das variáveis na representação

- Abordagem: **extensão das regras sintáticas** com conhecimento semântico

12

## SEMÂNTICA & SINTAXE

### ○ Regras estendidas

- $A \rightarrow \alpha_1 \dots \alpha_n \{f(\alpha_j.sem, \dots, \alpha_k.sem)\}$ 
  - Manipulação de atributos semânticos anexados às regras sintáticas
  - Função  $f$  que mapeia os atributos na representação de significado desejada para  $A$  ( $A.sem$ )

### ○ Duas opções mais sistemáticas

- Uso de notação lambda e lógica de 1ª ordem
- Estruturas de atributos e unificação

13

## 1ª ABORDAGEM: LÓGICA

### ○ Exemplo

- Sentença: *Maharani closed.*
- Fórmula lógica:  $\text{Closed}(\text{Maharani})$

### ○ Gramática

$S \rightarrow NP VP$   
 $NP \rightarrow \text{ProperNoun}$   
 $VP \rightarrow \text{Verb}$   
 $\text{ProperNoun} \rightarrow \text{Maharani}$   
 $\text{Verb} \rightarrow \text{closed}$

14

## 1ª ABORDAGEM: LÓGICA

### ○ Exemplo

- Sentença: *Maharani closed.*
- Fórmula lógica: Closed(Maharani)

### ○ Possível solução

ProperNoun → Maharani      {Maharani}  
 Gera-se a constante lógica Maharani, atribuída a  
 ProperNoun

15

## 1ª ABORDAGEM: LÓGICA

### ○ Exemplo

- Sentença: *Maharani closed.*
- Fórmula lógica: Closed(Maharani)

### ○ Possível solução

NP → ProperNoun      {ProperNoun.sem}  
 A especificação lógica de NP é a mesma de ProperNoun

16

## 1ª ABORDAGEM: LÓGICA

### ○ Exemplo

- Sentença: *Maharani closed.*
- Fórmula lógica: Closed(Maharani)

### ○ Possível solução

Verb  $\rightarrow$  closed  $\{\lambda x. \text{Closed}(x)\}$

A especificação lógica de Verb indica o nome do predicado, sua aridade (1, no caso) e a existência de um argumento que pode ser instanciado depois (no caso, com Maharani)

17

## 1ª ABORDAGEM: LÓGICA

### ○ Exemplo

- Sentença: *Maharani closed.*
- Fórmula lógica: Closed(Maharani)

### ○ Possível solução

VP  $\rightarrow$  Verb  $\{\text{Verb.sem}\}$

A especificação lógica de VP é a mesma de Verb

18

## 1ª ABORDAGEM: LÓGICA

### ○ Exemplo

- Sentença: *Maharani closed.*
- Fórmula lógica:  $\text{Closed}(\text{Maharani})$

### ○ Possível solução

$S \rightarrow \text{NP VP} \quad \{\text{VP.sem}(\text{NP.sem})\}$

A especificação de S indica que a fórmula lógica de NP deve instanciar a fórmula de VP (via redução lambda)

19

## 1ª ABORDAGEM: LÓGICA

### ○ Exemplo

- Sentença: *Maharani closed.*
- Fórmula lógica:  $\text{Closed}(\text{Maharani})$

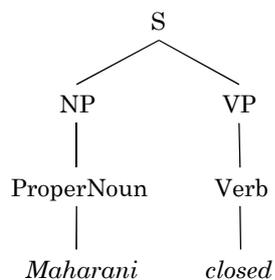
### ○ Possível solução

ProperNoun $\rightarrow$ Maharani	$\{\text{Maharani}\}$
NP $\rightarrow$ ProperNoun	$\{\text{ProperNoun.sem}\}$
Verb $\rightarrow$ closed	$\{\lambda x. \text{Closed}(x)\}$
VP $\rightarrow$ Verb	$\{\text{Verb.sem}\}$
S $\rightarrow$ NP VP	$\{\text{VP.sem}(\text{NP.sem})\}$

20

## 1ª ABORDAGEM: LÓGICA

## ○ Exemplo



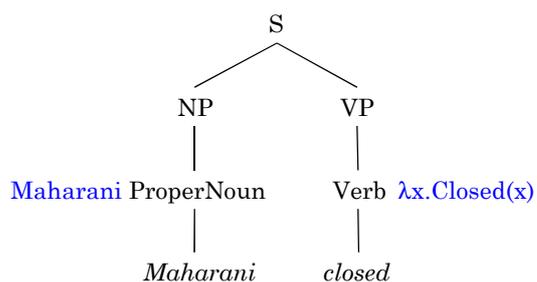
## Gramática estendida

ProperNoun $\rightarrow$ Maharani	{Maharani}
NP $\rightarrow$ ProperNoun	{ProperNoun.sem}
Verb $\rightarrow$ closed	{ $\lambda x.$ Closed(x)}
VP $\rightarrow$ Verb	{Verb.sem}
S $\rightarrow$ NP VP	{VP.sem(NP.sem)}

21

## 1ª ABORDAGEM: LÓGICA

## ○ Exemplo



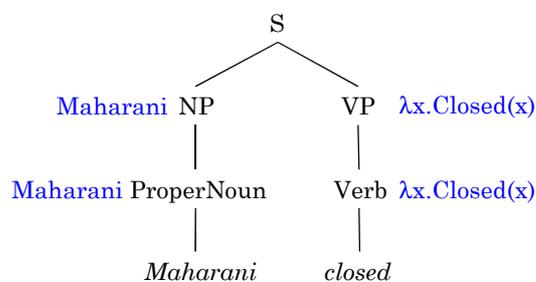
## Gramática estendida

ProperNoun $\rightarrow$ Maharani	{Maharani}
NP $\rightarrow$ ProperNoun	{ProperNoun.sem}
Verb $\rightarrow$ closed	{ $\lambda x.$ Closed(x)}
VP $\rightarrow$ Verb	{Verb.sem}
S $\rightarrow$ NP VP	{VP.sem(NP.sem)}

22

## 1ª ABORDAGEM: LÓGICA

## ○ Exemplo



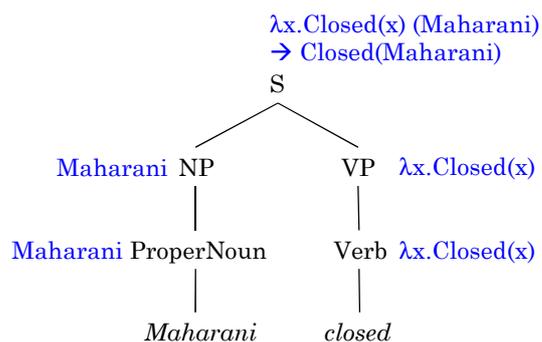
## Gramática estendida

ProperNoun $\rightarrow$ Maharani	{Maharani}
NP $\rightarrow$ ProperNoun	{ProperNoun.sem}
Verb $\rightarrow$ closed	{ $\lambda x.Closed(x)$ }
VP $\rightarrow$ Verb	{Verb.sem}
S $\rightarrow$ NP VP	{VP.sem(NP.sem)}

23

## 1ª ABORDAGEM: LÓGICA

## ○ Exemplo



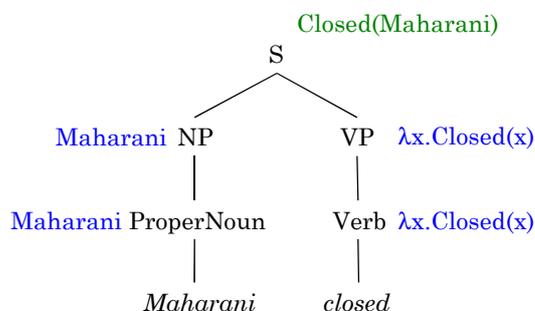
## Gramática estendida

ProperNoun $\rightarrow$ Maharani	{Maharani}
NP $\rightarrow$ ProperNoun	{ProperNoun.sem}
Verb $\rightarrow$ closed	{ $\lambda x.Closed(x)$ }
VP $\rightarrow$ Verb	{Verb.sem}
S $\rightarrow$ NP VP	{VP.sem(NP.sem)}

24

## 1ª ABORDAGEM: LÓGICA

## ○ Exemplo



## Gramática estendida

ProperNoun → Maharani	{Maharani}
NP → ProperNoun	{ProperNoun.sem}
Verb → closed	{λx.Closed(x)}
VP → Verb	{Verb.sem}
S → NP VP	{VP.sem(NP.sem)}

25

## 1ª ABORDAGEM: LÓGICA

## ○ Possíveis dificuldades

- Pode haver muita informação em nível lexical
- Cópia de valores de filhos para pais
- Uso de redução lambda

○ Deve-se pensar caso a caso, apesar do padrão de resolução se repetir○ Pode haver grande trabalho de “**engenharia gramatical**”

26

## 1ª ABORDAGEM: LÓGICA

- Exemplo um pouco mais complexo

- Discutir em duplas: como tratar a sentença abaixo?

Sentença: O homem lavou o carro.

Representação pretendida?

27

## 1ª ABORDAGEM: LÓGICA

- Exemplo um pouco mais complexo

- Discutir em duplas: como tratar a sentença abaixo?

Sentença: O homem lavou o carro.

Representação pretendida

Lavou(Homem,Carro)

Gramática? Extensão com atributos semânticos?

28

## 1ª ABORDAGEM: LÓGICA

○ Exemplo um pouco mais complexo

- Discutir em duplas: como tratar a sentença abaixo?

Sentença: O homem lavou o carro.

Representação pretendida

Lavou(Homem,Carro)

S → NP VP	{VP.sem(NP.sem)}
NP → Det Noun	{Noun.sem}
VP → Verb NP	{Verb.sem(NP.sem)}
Det → o	{}
Noun → carro	{Carro}
Noun → homem	{Homem}
Verb → lavou	{λx.λy.Lavou(y,x)}

Análise da sentença?

29

## 1ª ABORDAGEM: LÓGICA

○ Exemplo um pouco mais complexo

- Discutir em duplas: como tratar a sentença abaixo?

Sentença: *Every restaurant closed.*

Representação pretendida

$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)$

S → NP VP  
 NP → Det Nominal  
 VP → Verb  
 Nominal → Noun  
 Det → every  
 Noun → restaurant  
 Verb → closed

30

## 1ª ABORDAGEM: LÓGICA

### ○ Exemplo um pouco mais complexo

- **Discutir em duplas: como tratar a sentença abaixo?**

**Sentença:** *Every restaurant closed.*

**Representação pretendida**

$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)$

S → NP VP  
 NP → Det Nominal  
 VP → Verb  
 Nominal → Noun  
 Det → every  
 Noun → restaurant  
 Verb → closed

Quais as dificuldades?

31

## 1ª ABORDAGEM: LÓGICA

### ○ Exemplo

**Sentença:** *Every restaurant closed.*

**Representação pretendida**

$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)$

### ○ **Dificuldades**

- Quantificadores, conectivos
- Entrelaçamento do conteúdo do NP com o do VP

32

## 1ª ABORDAGEM: LÓGICA

○ Exemplo

Sentença: *Every restaurant closed.*

Representação pretendida

$$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)$$

## ○ Possível solução

Det  $\rightarrow$  every  $\quad \{\lambda P.\lambda Q.\forall x P(x) \Rightarrow Q(x)\}$

Permite-se que o lambda também atue sobre predicados, deixando a fórmula para *every* o mais genérica possível

33

## 1ª ABORDAGEM: LÓGICA

○ Exemplo

Sentença: *Every restaurant closed.*

Representação pretendida

$$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)$$

## ○ Possível solução

Noun  $\rightarrow$  restaurant  $\quad \{\lambda x.\text{Restaurant}(x)\}$

Gera-se um restaurante com um argumento a ser instanciado

34

## 1ª ABORDAGEM: LÓGICA

○ Exemplo

Sentença: *Every restaurant closed.*

Representação pretendida

$$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)$$

## ○ Possível solução

Nominal  $\rightarrow$  Noun {Noun.sem}

A especificação lógica de Nominal é a mesma de Noun

35

## 1ª ABORDAGEM: LÓGICA

○ Exemplo

Sentença: *Every restaurant closed.*

Representação pretendida

$$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)$$

## ○ Possível solução

NP  $\rightarrow$  Det Nominal {Det.sem(Nominal.sem)}

A especificação de NP indica que a fórmula lógica de Nominal deve instanciar a fórmula de Det (via redução lambda)

36

## 1ª ABORDAGEM: LÓGICA

○ Exemplo

Sentença: *Every restaurant closed.*

Representação pretendida

$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)$

## ○ Possível solução

Verb  $\rightarrow$  closed       $\{\lambda x. \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)\}$

A especificação de Verb indica uma fórmula lógica com o evento em si e um argumento a ser instanciado

37

## 1ª ABORDAGEM: LÓGICA

○ Exemplo

Sentença: *Every restaurant closed.*

Representação pretendida

$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)$

## ○ Possível solução

VP  $\rightarrow$  Verb       $\{\text{Verb.sem}\}$

A especificação lógica de VP é a mesma de Verb

38

## 1ª ABORDAGEM: LÓGICA

○ Exemplo

Sentença: *Every restaurant closed.*

Representação pretendida

$$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e, x)$$

## ○ Possível solução

$$S \rightarrow NP \text{ VP} \quad \{\text{NP.sem}(\text{VP.sem})\}$$

A especificação de S indica que a fórmula lógica de VP deve instanciar a fórmula de NP (via redução lambda)

39

## 1ª ABORDAGEM: LÓGICA

○ Exemplo

Sentença: *Every restaurant closed.*

Representação pretendida

$$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e, x)$$

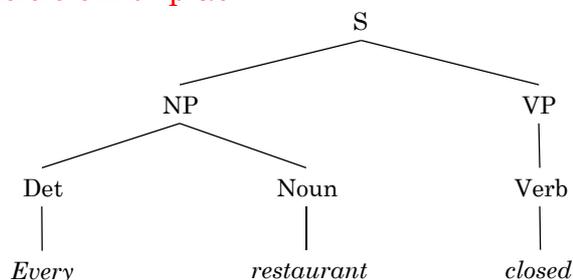
## ○ Possível solução

Det $\rightarrow$ every	$\{\lambda P. \lambda Q. \forall x P(x) \Rightarrow Q(x)\}$
Noun $\rightarrow$ restaurant	$\{\lambda x. \text{Restaurant}(x)\}$
Nominal $\rightarrow$ Noun	$\{\text{Noun.sem}\}$
NP $\rightarrow$ Det Nominal	$\{\text{Det.sem}(\text{Nominal.sem})\}$
Verb $\rightarrow$ closed	$\{\lambda x. \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e, x)\}$
VP $\rightarrow$ Verb	$\{\text{Verb.sem}\}$
S $\rightarrow$ NP VP	$\{\text{NP.sem}(\text{VP.sem})\}$

40

## 1ª ABORDAGEM: LÓGICA

## o Exercício em duplas

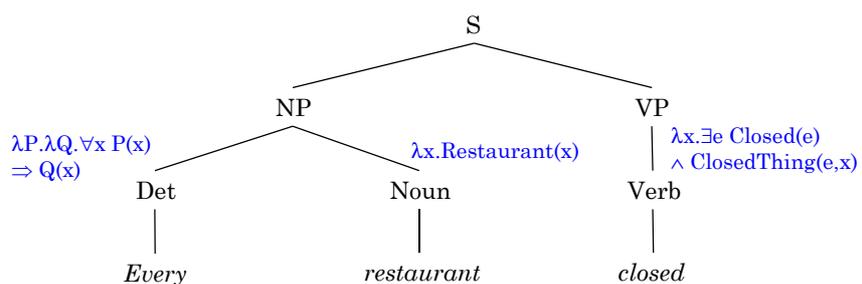


## Gramática estendida

Det → every	$\{\lambda P.\lambda Q.\forall x P(x) \Rightarrow Q(x)\}$
Noun → restaurant	$\{\lambda x.\text{Restaurant}(x)\}$
Nominal → Noun	$\{\text{Noun.sem}\}$
NP → Det Nominal	$\{\text{Det.sem}(\text{Nominal.sem})\}$
Verb → closed	$\{\lambda x.\exists e \text{Closed}(e) \wedge \text{ClosedThing}(e,x)\}$
VP → Verb	$\{\text{Verb.sem}\}$
S → NP VP	$\{\text{NP.sem}(\text{VP.sem})\}$

41

## 1ª ABORDAGEM: LÓGICA



## Gramática estendida

Det → every	$\{\lambda P.\lambda Q.\forall x P(x) \Rightarrow Q(x)\}$
Noun → restaurant	$\{\lambda x.\text{Restaurant}(x)\}$
Nominal → Noun	$\{\text{Noun.sem}\}$
NP → Det Nominal	$\{\text{Det.sem}(\text{Nominal.sem})\}$
Verb → closed	$\{\lambda x.\exists e \text{Closed}(e) \wedge \text{ClosedThing}(e,x)\}$
VP → Verb	$\{\text{Verb.sem}\}$
S → NP VP	$\{\text{NP.sem}(\text{VP.sem})\}$

42

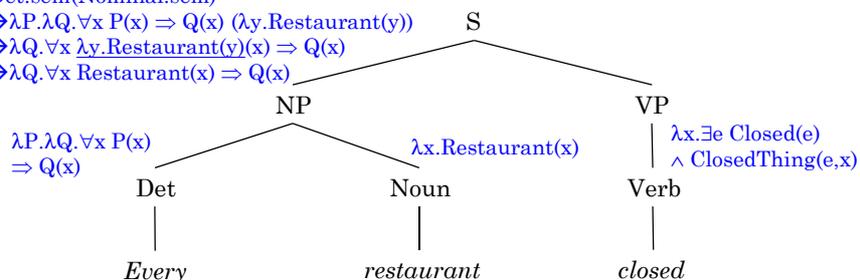
## 1ª ABORDAGEM: LÓGICA

Det.sem(Nominal.sem)

$\rightarrow \lambda P.\lambda Q.\forall x P(x) \Rightarrow Q(x)$  ( $\lambda y.$ Restaurant( $y$ ))

$\rightarrow \lambda Q.\forall x \lambda y.$ Restaurant( $y$ )( $x$ )  $\Rightarrow Q(x)$

$\rightarrow \lambda Q.\forall x$  Restaurant( $x$ )  $\Rightarrow Q(x)$

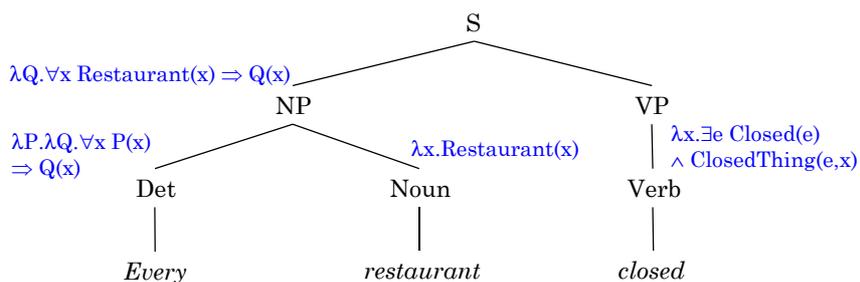


Gramática estendida

Det $\rightarrow$ every	$\{\lambda P.\lambda Q.\forall x P(x) \Rightarrow Q(x)\}$
Noun $\rightarrow$ restaurant	$\{\lambda x.$ Restaurant( $x$ ) $\}$
Nominal $\rightarrow$ Noun	$\{\text{Noun.sem}\}$
NP $\rightarrow$ Det Nominal	$\{\text{Det.sem}(\text{Nominal.sem})\}$
Verb $\rightarrow$ closed	$\{\lambda x.\exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)\}$
VP $\rightarrow$ Verb	$\{\text{Verb.sem}\}$
S $\rightarrow$ NP VP	$\{\text{NP.sem}(\text{VP.sem})\}$

43

## 1ª ABORDAGEM: LÓGICA

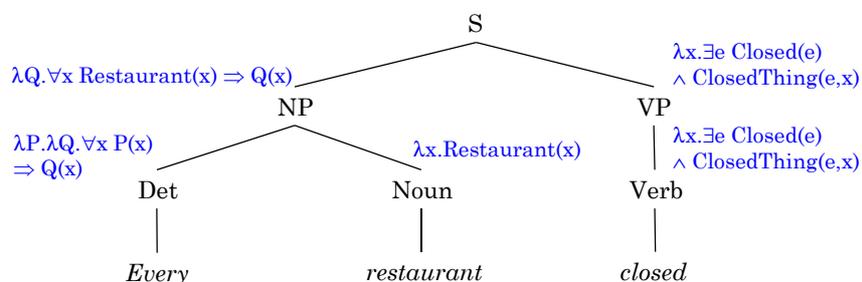


Gramática estendida

Det $\rightarrow$ every	$\{\lambda P.\lambda Q.\forall x P(x) \Rightarrow Q(x)\}$
Noun $\rightarrow$ restaurant	$\{\lambda x.$ Restaurant( $x$ ) $\}$
Nominal $\rightarrow$ Noun	$\{\text{Noun.sem}\}$
NP $\rightarrow$ Det Nominal	$\{\text{Det.sem}(\text{Nominal.sem})\}$
Verb $\rightarrow$ closed	$\{\lambda x.\exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)\}$
VP $\rightarrow$ Verb	$\{\text{Verb.sem}\}$
S $\rightarrow$ NP VP	$\{\text{NP.sem}(\text{VP.sem})\}$

44

## 1ª ABORDAGEM: LÓGICA

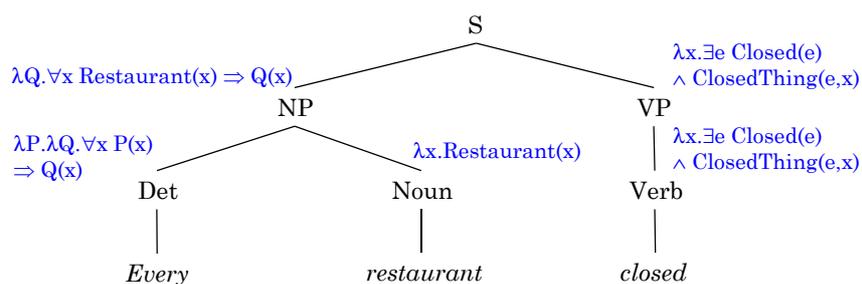


Gramática estendida

Det → every	{ $\lambda P.\lambda Q.\forall x P(x) \Rightarrow Q(x)$ }
Noun → restaurant	{ $\lambda x.\text{Restaurant}(x)$ }
Nominal → Noun	{Noun.sem}
NP → Det Nominal	{Det.sem(Nominal.sem)}
Verb → closed	{ $\lambda x.\exists e \text{Closed}(e) \wedge \text{ClosedThing}(e,x)$ }
VP → Verb	{Verb.sem}
S → NP VP	{NP.sem(VP.sem)}

45

NP.Sem(VP.sem)  
 $\rightarrow \lambda Q.\forall x \text{Restaurant}(x) \Rightarrow Q(x) (\lambda y.\exists e \text{Closed}(e) \wedge \text{ClosedThing}(e,y))$   
 $\rightarrow \forall x \text{Restaurant}(x) \Rightarrow \lambda y.\exists e \text{Closed}(e) \wedge \text{ClosedThing}(e,y)(x)$   
 $\rightarrow \forall x \text{Restaurant}(x) \Rightarrow \exists e \text{Closed}(e) \wedge \text{ClosedThing}(e,x)$

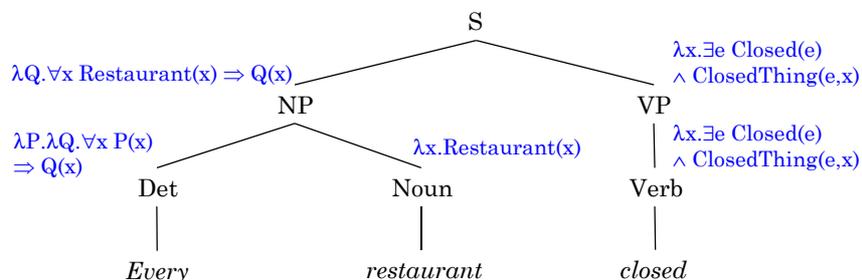


Gramática estendida

Det → every	{ $\lambda P.\lambda Q.\forall x P(x) \Rightarrow Q(x)$ }
Noun → restaurant	{ $\lambda x.\text{Restaurant}(x)$ }
Nominal → Noun	{Noun.sem}
NP → Det Nominal	{Det.sem(Nominal.sem)}
Verb → closed	{ $\lambda x.\exists e \text{Closed}(e) \wedge \text{ClosedThing}(e,x)$ }
VP → Verb	{Verb.sem}
S → NP VP	{NP.sem(VP.sem)}

46

## 1ª ABORDAGEM: LÓGICA

$$\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)$$


Gramática estendida

Det → every	{ $\lambda P.\lambda Q.\forall x P(x) \Rightarrow Q(x)$ }
Noun → restaurant	{ $\lambda x.\text{Restaurant}(x)$ }
Nominal → Noun	{Noun.sem}
NP → Det Nominal	{Det.sem(Nominal.sem)}
Verb → closed	{ $\lambda x.\exists e \text{ Closed}(e) \wedge \text{ClosedThing}(e,x)$ }
VP → Verb	{Verb.sem}
S → NP VP	{NP.sem(VP.sem)}

47

## 1ª ABORDAGEM: LÓGICA

- o Ambigüidade de escopo de quantificadores

- Sentença: *Every restaurante has a menu.*

- Possíveis representações

$$\forall x \text{ Restaurant}(x) \Rightarrow \exists y (\text{Menu}(y) \wedge \exists e (\text{Having}(e) \wedge \text{Haver}(e,x) \wedge \text{Had}(e,y)))$$

→ Todos os restaurantes têm menus

$$\exists y \text{ Menu}(y) \wedge \forall x (\text{Restaurant}(x) \Rightarrow \exists e (\text{Having}(e) \wedge \text{Haver}(e,x) \wedge \text{Had}(e,y)))$$

→ Existe um menu no mundo que é compartilhado por todos os restaurantes

48

## 1ª ABORDAGEM: LÓGICA

### ○ Ambigüidade de escopo de quantificadores

- Devemos ser capazes de **produzir todas as possibilidades**
  - Devemos saber que *restaurant* está no papel de *haver* e *menu* no de *had*, mas o resto fica em aberto
    - Subespecificação
- Sabemos que *has*, *every restaurant* e *a menu* nos fornecem

$$\exists e \text{ Having}(e) \wedge \text{Haver}(e,x) \wedge \text{Had}(e,y)$$

$$\forall x \text{ Restaurant}(x) \Rightarrow Q(x)$$

$$\exists x \text{ Menu}(x) \wedge Q(x)$$

49

## 1ª ABORDAGEM: LÓGICA

### ○ Solução: *Cooper storage* (Cooper, 1983)

- Armazenar as possibilidades junto às regras sintáticas
  - Indexam-se as possibilidades
  - Cada ordem de aplicação resulta em uma interpretação diferente

$$\exists e \text{ Having}(e) \wedge \text{Haver}(e,s1) \wedge \text{Had}(e,s2)$$

$$(\lambda Q. \forall x \text{ Restaurant}(x) \Rightarrow Q(x), 1)$$

$$(\lambda Q. \exists x \text{ Menu}(x) \wedge Q(x), 2)$$

- *Store*: representação principal + lista indexada de expressões quantificadas

50

## 1ª ABORDAGEM: LÓGICA

○ Solução: *Cooper storage* (Cooper, 1983)

- 1ª possibilidade de derivação

$$\begin{aligned} & \exists e \text{ Having}(e) \wedge \text{Haver}(e, s1) \wedge \text{Had}(e, s2) \\ & (\lambda Q. \forall x \text{ Restaurant}(x) \Rightarrow Q(x), 1) \\ & (\lambda Q. \exists x \text{ Menu}(x) \wedge Q(x), 2) \end{aligned}$$

$$\begin{aligned} & \lambda Q. \forall x (\text{Restaurant}(x) \Rightarrow Q(x)) (\lambda s1. \exists e \text{ Having}(e) \wedge \text{Haver}(e, s1) \wedge \text{Had}(e, s2)) \\ & \rightarrow \forall x (\text{Restaurant}(x) \Rightarrow (\lambda s1. \exists e \text{ Having}(e) \wedge \text{Haver}(e, s1) \wedge \text{Had}(e, s2)) (x)) \\ & \rightarrow \forall x (\text{Restaurant}(x) \Rightarrow \exists e \text{ Having}(e) \wedge \text{Haver}(e, x) \wedge \text{Had}(e, s2)) \end{aligned}$$

$$\begin{aligned} & \lambda Q. \exists y \text{ Menu}(y) \wedge Q(y) (\lambda s2. \forall x (\text{Restaurant}(x) \Rightarrow \exists e \text{ Having}(e) \wedge \text{Haver}(e, x) \wedge \text{Had}(e, s2))) \\ & \rightarrow \exists y \text{ Menu}(y) \wedge (\lambda s2. \forall x (\text{Restaurant}(x) \Rightarrow \exists e \text{ Having}(e) \wedge \text{Haver}(e, x) \wedge \text{Had}(e, s2))) (y) \\ & \rightarrow \exists y \text{ Menu}(y) \wedge \forall x (\text{Restaurant}(x) \Rightarrow \exists e \text{ Having}(e) \wedge \text{Haver}(e, x) \wedge \text{Had}(e, y)) \end{aligned}$$

51

## 1ª ABORDAGEM: LÓGICA

○ Solução: *Cooper storage* (Cooper, 1983)

- 2ª possibilidade de derivação

$$\begin{aligned} & \exists e \text{ Having}(e) \wedge \text{Haver}(e, s1) \wedge \text{Had}(e, s2) \\ & (\lambda Q. \forall x \text{ Restaurant}(x) \Rightarrow Q(x), 1) \\ & (\lambda Q. \exists x \text{ Menu}(x) \wedge Q(x), 2) \end{aligned}$$

$$\begin{aligned} & \lambda Q. \exists x (\text{Menu}(x) \wedge Q(x)) (\lambda s2. \exists e \text{ Having}(e) \wedge \text{Haver}(e, s1) \wedge \text{Had}(e, s2)) \\ & \rightarrow \exists x (\text{Menu}(x) \wedge (\lambda s2. \exists e \text{ Having}(e) \wedge \text{Haver}(e, s1) \wedge \text{Had}(e, s2))(x)) \\ & \rightarrow \exists x (\text{Menu}(x) \wedge \exists e \text{ Having}(e) \wedge \text{Haver}(e, s1) \wedge \text{Had}(e, x)) \end{aligned}$$

$$\begin{aligned} & \lambda Q. \forall x (\text{Restaurant}(x) \Rightarrow Q(x)) (\lambda s1. \exists y (\text{Menu}(y) \wedge \exists e \text{ Having}(e) \wedge \text{Haver}(e, s1) \wedge \text{Had}(e, y))) \\ & \rightarrow \forall x (\text{Restaurant}(x) \Rightarrow (\lambda s1. \exists y (\text{Menu}(y) \wedge \exists e \text{ Having}(e) \wedge \text{Haver}(e, s1) \wedge \text{Had}(e, y)))(x)) \\ & \rightarrow \forall x \text{ Restaurant}(x) \Rightarrow \exists y (\text{Menu}(y) \wedge \exists e \text{ Having}(e) \wedge \text{Haver}(e, x) \wedge \text{Had}(e, y)) \end{aligned}$$

52

## 1ª ABORDAGEM: LÓGICA

### ○ Solução: *Cooper storage* (Cooper, 1983)

- **Desvantagens**

- Para cada tipo de ambigüidade, mais sofisticada e ad hoc fica a representação
- Por exemplo, como lidar com negação?

Sentença: *Every restaurant did not close.*

Representações possíveis

$\neg(\forall x \text{ Restaurant}(x) \Rightarrow \exists e \text{ Closing}(e) \wedge \text{Closed}(e,x))$

$\forall x \text{ Restaurant}(x) \Rightarrow \neg(\exists e \text{ Closing}(e) \wedge \text{Closed}(e,x))$

- Não aceita restrições adicionais que podem ser interessantes

53

## 1ª ABORDAGEM: LÓGICA

### ○ Outra solução: **abordagem por restrições**

- *Hole semantics* (Bos, 1996)

- Inicialmente, determinam-se slots (*holes*) que devem ser preenchidos nas expressões lógicas (alternativa para a redução lambda)

$\forall x \text{ Restaurant}(x) \Rightarrow \text{hole1}$

$\exists x \text{ Menu}(x) \wedge \text{hole2}$

$\exists e \text{ Having}(e) \wedge \text{Haver}(e,x) \wedge \text{Had}(e,y)$

54

## 1ª ABORDAGEM: LÓGICA

### ○ Outra solução: abordagem por restrições

- *Hole semantics* (Bos, 1996)
  - Para referência, rotulam-se as expressões

R1:  $\forall x \text{ Restaurant}(x) \Rightarrow \text{hole1}$

R2:  $\exists x \text{ Menu}(x) \wedge \text{hole2}$

R3:  $\exists e \text{ Having}(e) \wedge \text{Haver}(e,x) \wedge \text{Had}(e,y)$

55

## 1ª ABORDAGEM: LÓGICA

### ○ Outra solução: abordagem por restrições

- *Hole semantics* (Bos, 1996)
  - Determinam-se restrições (de dominância) sobre que expressões podem preencher que slots

R1:  $\forall x \text{ Restaurant}(x) \Rightarrow \text{hole1}$

R2:  $\exists x \text{ Menu}(x) \wedge \text{hole2}$

R3:  $\exists e \text{ Having}(e) \wedge \text{Haver}(e,x) \wedge \text{Had}(e,y)$

#### Restrições de dominância:

Inicialmente, tanto R1 quanto R2 podem iniciar o processo (preenchendo um hole0 genérico hipotético)

hole1 pode ser preenchido com (domina) R2 ou hole2 pode ser preenchido com (domina) R1

hole1 pode ser preenchido com (domina) R3

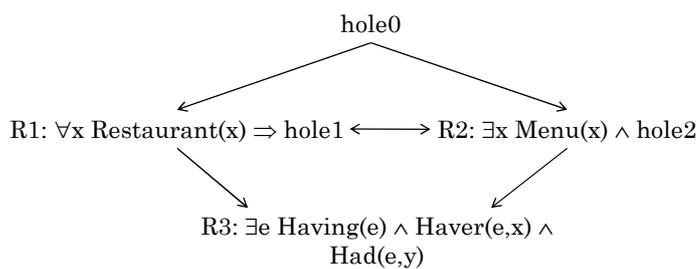
hole2 pode ser preenchido com (domina) R3

56

## 1ª ABORDAGEM: LÓGICA

## ○ Outra solução: abordagem por restrições

- *Hole semantics* (Bos, 1996)

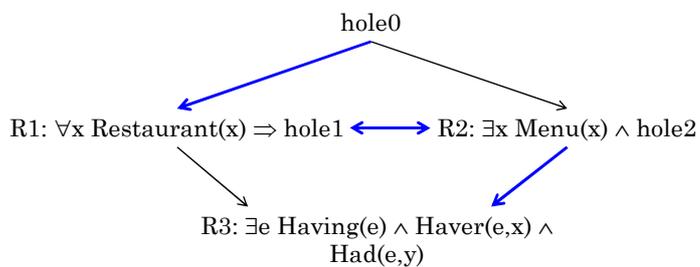
○ Exemplo

57

## 1ª ABORDAGEM: LÓGICA

## ○ Outra solução: abordagem por restrições

- *Hole semantics* (Bos, 1996)

○ Exemplo: possibilidade de interpretação

58

## 2ª ABORDAGEM: ESTRUTURAS DE ATRIBUTOS

### ○ Estruturas de atributos e unificação

- Recursos tão poderosos que também podem representar semântica
- Exemplo: *Rhumba closed*.
  - Em lógica de 1ª ordem:  $\exists e \text{ Closing}(e) \wedge \text{Closed}(e, \text{Rhumba})$

- Em estrutura de atributos:
 

Quant	$\exists$																
Var	$\boxed{1}$																
Formula	<table style="border-collapse: collapse;"> <tr> <td style="padding-right: 5px;">Op</td> <td style="padding-left: 5px;"><math>\wedge</math></td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">Formula1</td> <td style="padding-left: 5px;"> <table style="border-collapse: collapse;"> <tr> <td style="padding-right: 5px;">Pred</td> <td style="padding-left: 5px;">Closing</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">Arg0</td> <td style="padding-left: 5px;"><math>\boxed{1}</math></td> </tr> </table> </td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">Formula2</td> <td style="padding-left: 5px;"> <table style="border-collapse: collapse;"> <tr> <td style="padding-right: 5px;">Pred</td> <td style="padding-left: 5px;">Closed</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">Arg0</td> <td style="padding-left: 5px;"><math>\boxed{1}</math></td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">Arg1</td> <td style="padding-left: 5px;">Rhumba</td> </tr> </table> </td> </tr> </table>	Op	$\wedge$	Formula1	<table style="border-collapse: collapse;"> <tr> <td style="padding-right: 5px;">Pred</td> <td style="padding-left: 5px;">Closing</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">Arg0</td> <td style="padding-left: 5px;"><math>\boxed{1}</math></td> </tr> </table>	Pred	Closing	Arg0	$\boxed{1}$	Formula2	<table style="border-collapse: collapse;"> <tr> <td style="padding-right: 5px;">Pred</td> <td style="padding-left: 5px;">Closed</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">Arg0</td> <td style="padding-left: 5px;"><math>\boxed{1}</math></td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">Arg1</td> <td style="padding-left: 5px;">Rhumba</td> </tr> </table>	Pred	Closed	Arg0	$\boxed{1}$	Arg1	Rhumba
Op	$\wedge$																
Formula1	<table style="border-collapse: collapse;"> <tr> <td style="padding-right: 5px;">Pred</td> <td style="padding-left: 5px;">Closing</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">Arg0</td> <td style="padding-left: 5px;"><math>\boxed{1}</math></td> </tr> </table>	Pred	Closing	Arg0	$\boxed{1}$												
Pred	Closing																
Arg0	$\boxed{1}$																
Formula2	<table style="border-collapse: collapse;"> <tr> <td style="padding-right: 5px;">Pred</td> <td style="padding-left: 5px;">Closed</td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">Arg0</td> <td style="padding-left: 5px;"><math>\boxed{1}</math></td> </tr> <tr> <td style="border-right: 1px solid black; padding-right: 5px;">Arg1</td> <td style="padding-left: 5px;">Rhumba</td> </tr> </table>	Pred	Closed	Arg0	$\boxed{1}$	Arg1	Rhumba										
Pred	Closed																
Arg0	$\boxed{1}$																
Arg1	Rhumba																

59

## 2ª ABORDAGEM: ESTRUTURAS DE ATRIBUTOS

### ○ Estruturas de atributos e unificação

- Produção pela associação de especificações às regras sintáticas

#### **S → NP VP**

$\langle S \text{ sem} \rangle = \langle NP \text{ sem} \rangle$   
 $\langle NP \text{ escopo} \rangle = \langle VP \text{ sem} \rangle$   
 $\langle VP \text{ arg0} \rangle = \langle NP \text{ var} \rangle$

#### **NP → ProperNoun**

$\langle NP \text{ sem} \rangle = \langle \text{ProperNoun sem} \rangle$   
 $\langle NP \text{ var} \rangle = \langle \text{ProperNoun var} \rangle$   
 $\langle NP \text{ escopo} \rangle = \langle \text{ProperNoun escopo} \rangle$

#### **ProperNoun → Rhumba**

$\langle \text{ProperNoun sem pred} \rangle = \text{Rhumba}$   
 $\langle \text{ProperNoun var} \rangle = \langle \text{ProperNoun sem pred} \rangle$

60

## 2ª ABORDAGEM: ESTRUTURAS DE ATRIBUTOS

- Estruturas de atributos e unificação
  - Produção pela associação de especificações às regras sintáticas

### VP → Verb

⟨VP sem⟩ = ⟨Verb sem⟩  
 ⟨VP arg0⟩ = ⟨Verb arg0⟩

### Verb → closed

⟨Verb sem quant⟩ = ∃  
 ⟨Verb sem formula op⟩ = ∧  
 ⟨Verb sem formula formula1 pred⟩ = Closing  
 ⟨Verb sem formula formula1 arg0⟩ = ⟨Verb sem var⟩  
 ⟨Verb sem formula formula2 pred⟩ = Closed  
 ⟨Verb sem formula formula2 arg0⟩ = ⟨Verb sem var⟩  
 ⟨Verb sem formula formula2 arg1⟩ = ⟨Verb arg0⟩

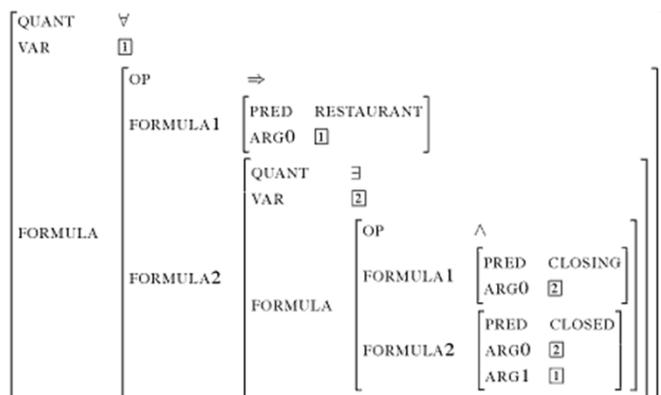
61

## 2ª ABORDAGEM: ESTRUTURAS DE ATRIBUTOS

- Pode se ter estruturas tão complexas quanto se queira

Every restaurant closed.

$\forall x \text{Restaurant}(x) \Rightarrow (\exists e \text{Closing}(e) \wedge \text{Closed}(e,x))$



62

## COMPOSICIONALIDADE

### ○ Não reflete toda a língua

- Há muitos fenômenos que não obedecem a composicionalidade

#### ○ Exemplos

- *Esse problema é só a ponta do iceberg.*
- *Está chovendo canivete.*

63

## COMPOSICIONALIDADE

### ○ Possível tratamento na gramática

- Listagem exaustiva dos possíveis significados

SN → a ponta do iceberg      {começo}

64

## COMPOSICIONALIDADE

### ○ Possível tratamento na gramática

- Listagem exaustiva dos possíveis significados

SN → a ponta do iceberg      {começo}

- Sempre há outras possibilidades

*Esse problema é só a **maldita** ponta do iceberg.*

*Esse problema é só a **maldita** ponta **de um** iceberg **enorme**.*

SN → a PONTA<sub>SN</sub> prep ICEBERG<sub>SN</sub>

PONTA<sub>SN</sub> → adj ponta

...

65

## ANÁLISE SEMÂNTICA SEM SINTAXE

### ○ **Nem toda** análise semântica precisa ser dirigida pela sintaxe

- Por exemplo, tentativas de Schank com a Teoria da Dependência Conceitual
  - Muita informação e procedimentos associados aos itens lexicais

66

## MÉTODOS DE ANÁLISE SEMÂNTICA

- Métodos anteriores não necessariamente precisam produzir fórmulas lógicas
- Qualquer representação suficientemente bem formalizada/entendida poderia ser usada
  - A escolha depende da necessidade