Representing Meaning Part 4 & Semantic Analysis

ICS 482 Natural Language Processing

Lecture 21: Representing Meaning Part 4 & Semantic Analysis

Husni Al-Muhtaseb

بسم الله الرحمن الرحيم

ICS 482 Natural Language Processing

Lecture 21: Representing Meaning Part 4 & Semantic Analysis

Husni Al-Muhtaseb

NLP Credits and Acknowledgment

These slides were adapted from presentations of the Authors of the book

---

SPEECH and LANGUAGE PROCESSING:

An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition

and some modifications from presentations found in the WEB by several scholars including the following

NLP Credits and Acknowledgment

If your name is missing please contact me
muhtaseb
At
Kfupm.
Edu.
sa

NLP Credits and Acknowledgment

Husni Al-Muhtaseb
James Martin
Jim Martin
Dan Jurafsky
Sandiway Fong
Song young in
Paula Matuszek
Mary-Angela Papalaskari
Dick Crouch
Tracy Kin
L. Venkata Subramaniam
Martin Volk
Bruce R. Maxim
Jan Hajič
Srinath Srinivasa
Simeon Ntafos
Paolo Pirjanian
Ricardo Vilalta
Tom Lenaerts

Heshaam Feili
Björn Gambäck
Christian Korthals
Thomas G. Dietterich
Devika Subramanian
Duminda Wijesekera
Lee McCluskey
David J. Kriegman
Kathleen McKeown
Michael J. Ciaraldi
David Finkel
Min-Yen Kan
Andreas Geyer-Schulz
Franz J. Kurfess
Tim Finin
Nadjet Bouayad
Kathy McCoy
Hans Uszkoreit
Azadeh Maghsoodi

Khurshid Ahmad
Staffan Larsson
Robert Wilensky
Feiyu Xu
Jakub Piskorski
Rohini Srihari
Mark Sanderson
Andrew Elks
Marc Davis
Ray Larson
Jimmy Lin
Marti Hearst
Andrew McCallum
Nick Kushmerick
Mark Craven
Chia-Hui Chang
Diana Maynard
James Allan

Martha Palmer
julia hirschberg
Elaine Rich
Christof Monz
Bonnie J. Dorr
Nizar Habash
Massimo Poesio
David Goss-Grubbs
Thomas K Harris
John Hutchins
Alexandros
Potamianos
Mike Rosner
Latifa Al-Sulaiti
Giorgio Satta
Jerry R. Hobbs
Christopher Manning
Hinrich Schütze
Alexander Gelbukh
Gina-Anne Levow
Guitao Gao
Qing Ma
Zeynep Altan

Previous Lectures

- Introduction and Phases of an NLP system
- NLP Applications - Chatting with Alice
- Finite State Automata & Regular Expressions & languages
- Morphology: Inflectional & Derivational
- Parsing and Finite State Transducers, Porter Stemmer
- Statistical NLP – Language Modeling
- N Grams, Smoothing
- Parts of Speech - Arabic Parts of Speech
- Syntax: Context Free Grammar (CFG) & Parsing
- Parsing: Earley’s Algorithm
- Probabilistic Parsing
- Probabilistic CYK - Dependency Grammar
- Semantics: Representing meaning
- Semantics: FOPC
- Lexicons and Morphology – invited lecture
- Semantics: Representing meaning

Today's Lecture

- Administration
  - Presentations Schedule
  - Teams for project (2 each)
  - Projects
- Lecture
  - Representing Meaning Part 4
  - Semantic Analysis

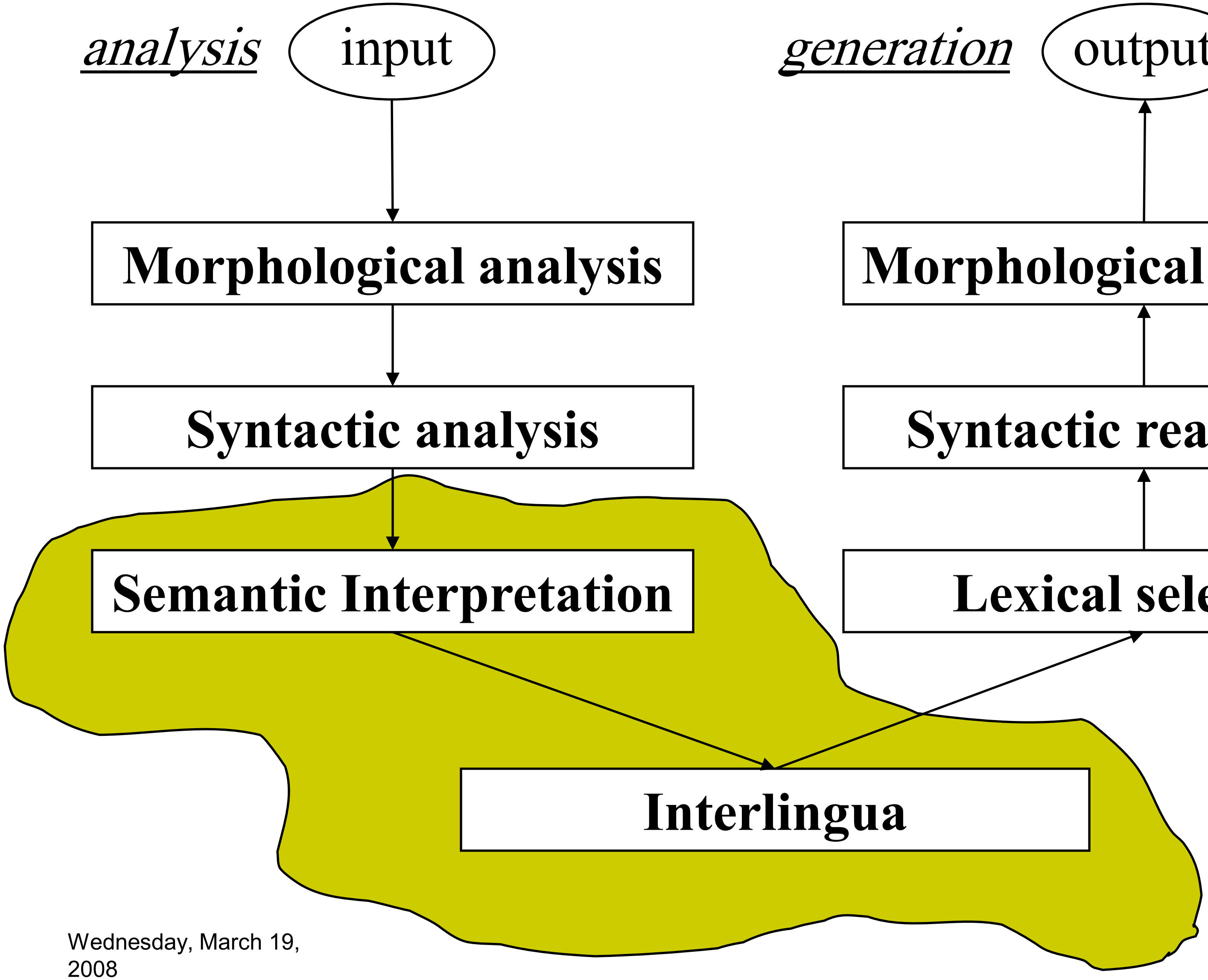
Presentations Schedule

- Presentations at class time
- 13th, 15th, 20th, and 22nd May
- visit the calendar section of this website
- Go to the month of May
- choose one slot in one of the assigned days for presentations
- Add a public entry in the most suitable slot for you
- Max 3 students per slot
- Presentation time: 25 minutes
  - 20 for presentation
  - 5 for discussions
- Put the title of your topic in the entry you are adding

Team

- 2-3 Members (alone)
  - Team Name (Your own)
  - Team logo (Your design idea)
  - By next class
- How to choose Team members
  - Similar goal
  - Easiness of communications
  - Consistency, harmony, and relaxation
  - ??
- WebCt Discussion list – Team Selection
- Project Ideas?

Machine Translation



Representation of Categories

- Categories are sets of objects or relations where all members share a set of features
- Method 1:
  - Create a unary predicate for each category
    - *VegetarianRestaurant(Maharani)*
  - Problem: Unable to talk about *VegetarianRestaurant*
    - Not a valid FOPC formula:
      - *MostPopular(Maharani, VegetarianRestaurant)*

Representation of Categories

- Method 2:
  - Reification اعتبره شيئا ماديا: Represent all concepts that we want to make statements about as full-fledged objects
  - *isa(Maharani, VegetarianRestaurant)*
  - *ako(VegetarianRestaurant, Restaurant)*
    (a kind of)

  - Reification: To regard or treat (an abstraction) as if it had concrete or material existence.

    www.dictionary.com

Representation of Events

- Not always single predicate
  - *I ate*
  - *I ate a turkey sandwich*
  - *I ate a turkey sandwich at my desk*
  - *I ate at my desk*
  - *I ate lunch*
  - *I ate a turkey sandwich for lunch*
  - *I ate a turkey sandwich for lunch at my desk*

Representation of Events

- Method 1:
  - –Create as many *different* eating predicates as are needed to handle all of the ways that eat behaves
  - *Eating1(Speaker)*
  - *Eating2(Speaker, TurkeySandwich)*
  - *Eating3(Speaker, TurkeySandwich, Desk)*
  - *Eating4(Speaker, Desk)*
  - *Eating5(Speaker, Lunch)*
  - *Eating6(Speaker, TurkeySandwich, Lunch)*
  - *Eating7(Speaker, TurkeySandwich, Lunch, Desk)*
  - Relate them using meaning postulates:
    - $\forall w, x, y, z\ Eating7(w, x, y, z) \Rightarrow Eating6(w, x, y)$

Representation of Events

- Problems:
  - Need too many meaning postulates
  - Difficult to scale up
- Method 2:
  - Use a *single* predicate where as many arguments are included in the definition of the predicate as ever appear with it in an input

Representation of Events

- $\exists$ *w, x, y Eating(Speaker, w, x, y)*
- $\exists$ *w, x Eating(Speaker, TurkeySandwich, w, x)*
- $\exists$ *w Eating(Speaker, TurkeySandwich, w, Desk)*
- $\exists$ *w, x Eating(Speaker, w, x, Desk)*
- $\exists$ *w, x Eating(Speaker, w, Lunch, x)*
- $\exists$ *w Eating(Speaker, TurkeySandwich, Lunch, w)*
- *Eating(Speaker, TurkeySandwich, Lunch, Desk)*

Representation of Events

- Problems:
  - Make too many commitments
    - Need to commit to all arguments (e.g., every eating event must be associated with a meal, which is not true)
  - Unable to refer to individual events
    - Event is a predicate, not a term

Representation of Events

- Method 3:
  - Use reification to elevate events to objects
  - Arguments of an event appear as predicates
  - Do not need to commit to arguments (roles) not mentioned in the input
  - Meaning postulates not needed

Representation of Events

- I ate.
  - $\exists$ w *isa(w, Eating)* $\wedge$ *Eater(w, Speaker)*
- I ate a turkey sandwich.
  - $\exists$ *w isa(w, Eating)* $\wedge$ *Eater(w, Speaker)* $\wedge$ *Eaten(w, TurkeySandwich)*
- I ate a turkey sandwich for lunch.
  - $\exists$ *w isa(w, Eating)* $\wedge$ *Eater(w, Speaker)* $\wedge$ *Eaten(w, TurkeySandwich)* $\wedge$ *MealEaten(w, Lunch)*

Temporal Representations

- How do we represent time and temporal relationships between events?
  - *Last year Ali was happy but soon he will be sad.*
- Where do we get temporal information?
  - Verb tense
  - Temporal expressions
  - Sequence of presentation

Temporal Representations

- I arrived in New York.
  - $\exists$ *i, e, w isa(w, Arriving)* $\wedge$ *Arriver(w, Speaker)* $\wedge$ *Destination(w, NewYork)* $\wedge$ *IntervalOf(w, i)* $\wedge$ *EndPoint(i, e)* $\wedge$ *Precedes(e, Now)*
- I am arriving in New York.
  - $\exists$ *i, w isa(w, Arriving)* $\wedge$ *Arriver(w, Speaker)* $\wedge$ *Destination(w, NewYork)* $\wedge$ *IntervalOf(w, i)* $\wedge$ *MemberOf(i, Now)*
- I will arrive in New York.
  - $\exists$ *i, b, w isa(w, Arriving)* $\wedge$ *Arriver(w, Speaker)* $\wedge$ *Destination(w, NewYork)* $\wedge$ *IntervalOf(w, i)* $\wedge$ *BeginPoint(i, b)* $\wedge$ *Precedes(Now, b)*

Representations of Time (cont.)

- The relation between simple verb tenses and points in time is not straightforward.
  - We fly from Dhahran to Riyadh.
    - present tense refers to a future event
  - Flight 12 will be at gate an hour now.
    - future tense refers to a past event
- In some formalisms, the tense of a sentence is expressed with the relation among ***times of events (E)*** in that sentence, ***time of a reference point (R)***, and ***time of utterance (U)***.

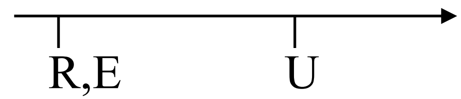
Reinhenbach's Approach to Representing Tenses

Past Perfect

I had eaten.

Past

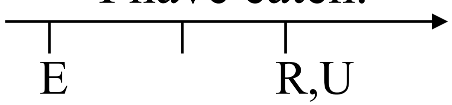
I ate.



- ***times of events (E)***
- ***time of a reference point (R)***
- ***time of utterance (U)***

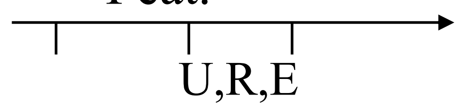
Present Perfect

I have eaten.



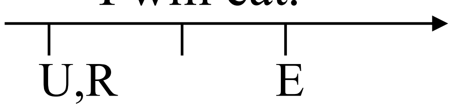
Present

I eat.



Future

I will eat.



Future Perfect

I will have eaten.

Aspects representation

- Aspects indicate relations between predicates and time.
  - Whether an event is conceptualized as happening at a point in time or over some time interval
  - Whether an event has ended or is ongoing
  - Whether any particular state comes about because of an event
- Event expressions: (based on the above)
  - Statives
  - Activities
  - Accomplishments
  - Achievements

Verbs and Event Types: Aspect

- **Statives**: states or properties of objects at a particular point in time
  - *Ali needs sleep.*
  - **Ali is needing sleep.*
  - **Need sleep.*
  - **Mary needs sleep in a week.*

•Odd When used in

•Progressive

•Imperative

•Not easily modified by adverbs like carefully

•Odd when modified with temporal expression using *in*

- **Activities**: events with no clear endpoint
  - *Ali drives a Porsche.*
  - **Ali drives a Porsche in a week.*

Verbs and Event Types: Aspect

- **Accomplishments**: events with durations and endpoints that result in some change of state
  - *Sami filled out the form.*
  - *Sami stopped filling out the form (Sami did not fill out the form) vs. Ali stopped driving a Porsche (Ali still drove a Porsche …for a while)*
- **Achievements**: events that change state but have no particular duration
  - *Asem reached the top.*
  - ** Asem stopped reaching the top.*
  - ** Asem reached the top for a few minutes.*

Aspects

- A verb can belong to different aspectual classes depending on the context
  - *I flew*. (activity)
  - *I flew to New York*. (accomplishment)

Beliefs

- In addition to the "real world", there are hypothetical world. Words such as: *believe, want, imagine, know* create speaker's hypothetical world.

I believe that Asem ate Saudi food.

FOPC: $\exists$ *u, v isa(u, believing)* $\wedge$ *isa(v, eating)* $\wedge$ *believer(u, speaker)* $\wedge$ *believedProp(u, v)* $\wedge$ *eater(v, Asem)* $\wedge$ *eaten(v, Saudifood)*

- There is a problem: every part of the above has to be true in order for the whole formula to be true
- But we don't know if this is true or not.

Beliefs

- How about a representation like this:

believing(Speaker, Eating(Asem , Saudifood))

- This is not FOPC, second argument is a formula not a term
- Predicates in FOPC hold between the objects in the domain being modeled, not between the relations that hold among the objects in the domain. Therefore, FOPC lacks a meaningful way to assert relations about full propositions.

Solution

- Introduce an operator, called *believes*, that takes two FOPC formulas as arguments, a formula for *believer*, and a formula for *believed proposition*.

Believes (speaker, $\exists$ *v isa (v, eating)* $\wedge$ *eater (v, Asem)* $\wedge$ *eaten (v, Saudifood))*

  - *Not FOPC*
  - Modal operators: *believe, want, imagine, know*
  - Modal Logic: is a logic augmented with modal operators

Frames

- We may use other representation languages instead of FOPC. But they will be equivalent to their representations in FOPC.
- For example, we may use **frames** to represent our *believing* example.

$$
\begin{bmatrix}
\text{BELIEVING} & \\
\text{BELIEVER} & \textit{Speaker} \\
\text{BELIEVED} & \begin{bmatrix} \text{EATING} & \\ \text{EATER} & \textit{Asem} \\ \text{EATEN} & \textit{SaudiFood} \end{bmatrix}
\end{bmatrix}
$$

بسم الله الرحمن الرحيم

ICS 482 Natural Language Processing

Lecture 21: Semantic Analysis

Husni Al-Muhtaseb

Semantic Analysis

- Semantic analysis is the process of taking in some linguistic input and assigning a meaning representation to it.
  - Different ways
    - make use of syntax
  - We're going to start with the idea that syntax does matter
    - The compositional rule-to-rule approach

Compositional Analysis

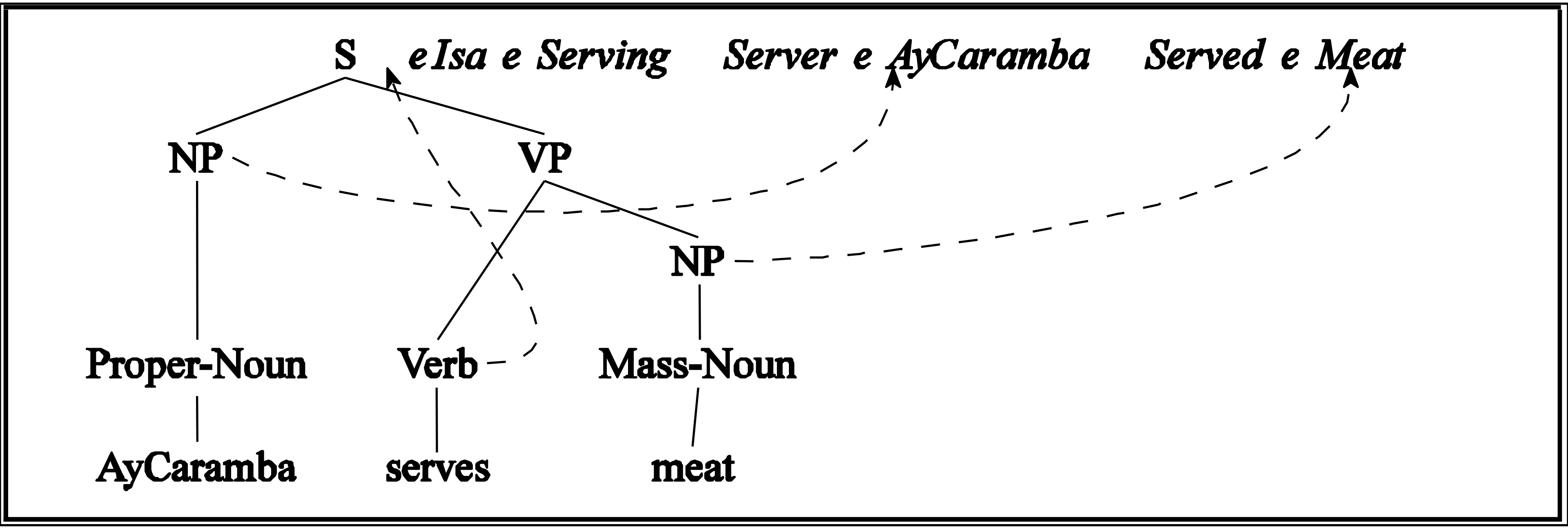
- Principle of Compositionality
  - The meaning of a whole is derived from the meanings of the parts
- What parts?
  - Account for the meaning not solely for the words, but also Syntactic Components and Relations
    - Ordering
    - Grouping
    - Relations among the words
  - The constituents of the syntactic parse of the input

Example

- AyCaramba serves meat

$$\exists e\ Serving(e) \wedge Server(e, AyCaramba) \wedge Served(e, Meat)$$

Compositional Analysis



Augmented Rules

- We'll accomplish this by attaching semantic formation rules to our syntactic CFG rules
- Abstractly

$$A \rightarrow \alpha_1 \ldots \alpha_n \qquad \{f(\alpha_1.sem, \ldots \alpha_n.sem)\}$$

- This should be read as the semantics we attach to A can be computed from some function applied to the semantics of A's parts.

Example

- Easy parts…
  - NP → PropNoun
  - NP → MassNoun
  - PropNoun → AyCaramba
  - MassMoun → meat
- Attachments

  {PropNoun.sem}

  {MassNoun.sem}

  {AyCaramba}

  {meat}

- These attachments consist of assigning constants and copying from daughters up to mothers.

Example

- $S \rightarrow NP\ VP$
- $VP \rightarrow Verb\ NP$
- $Verb \rightarrow serves$

- {VP.sem (NP.sem)}
- {Verb.sem (NP.sem)}

$$\lambda x\, \lambda y\ \exists e\ Serving(e) \wedge Server(e,y) \wedge Served(e,x)$$

These consist of taking the semantics attached to one daughter and applying it as a function to the semantics of the other daughters.

Lambda Forms

- A simple addition to FOPC
  - Take an FOPC sentence with variables in it that are to be bound.
  - Allow those variables to be bound by treating the lambda form as a function with formal arguments

$$\lambda x P(x)$$

$$\lambda x P(x)(Sally)$$

$$P(Sally)$$

Lambda Forms

$$\lambda x\, \lambda y In(x, y) \wedge Country(y)$$

$$\lambda x\, \lambda y In(x, y) \wedge Country(y)(BC)$$

$$\lambda y In(BC, y) \wedge Country(y)$$

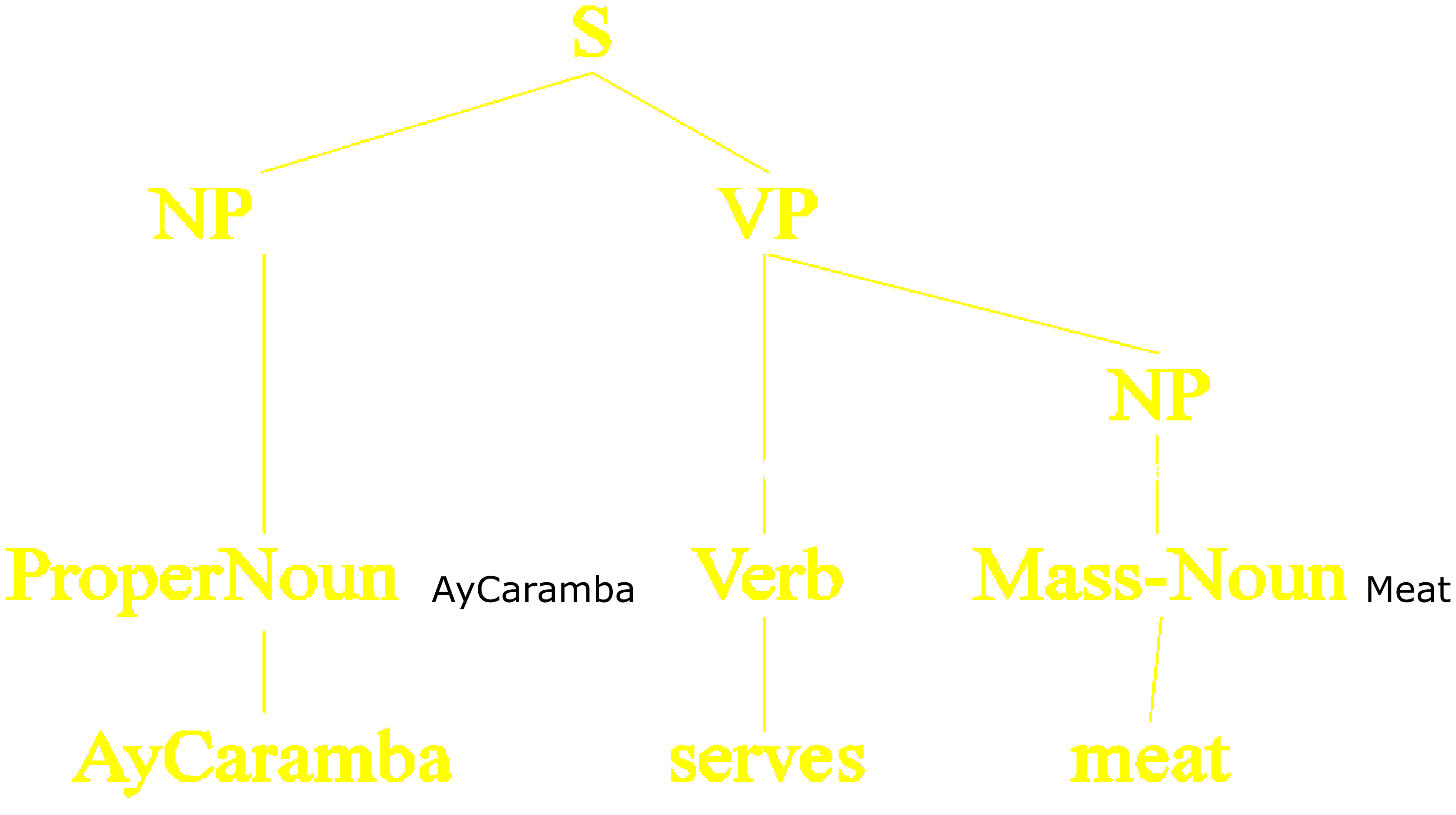
$$\lambda y In(BC, y) \wedge Country(y)$$

$$\lambda y In(BC, y) \wedge Country(y)(CANADA)$$

$$In(BC, CANADA) \wedge Country(CANADA)$$

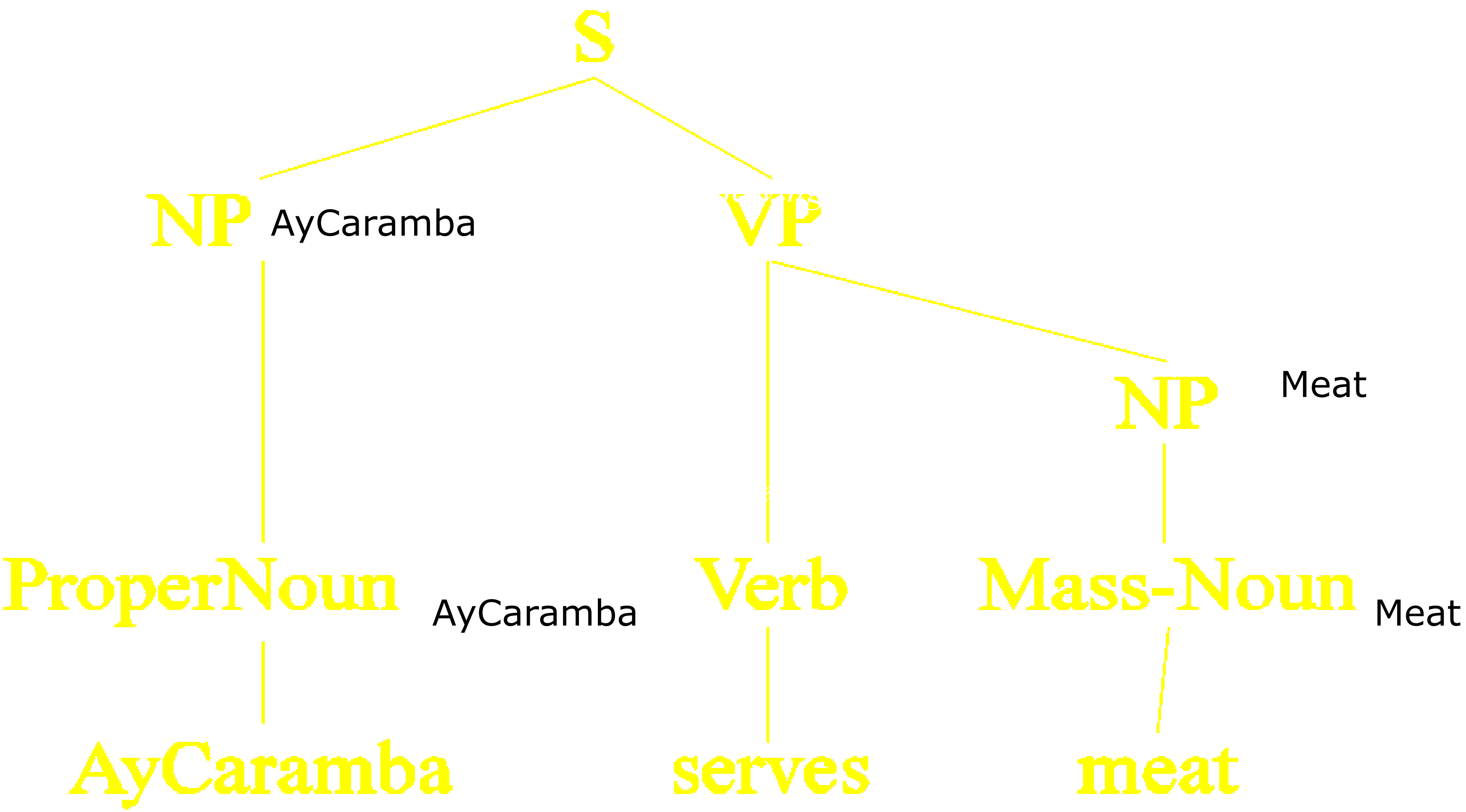
Example

Example



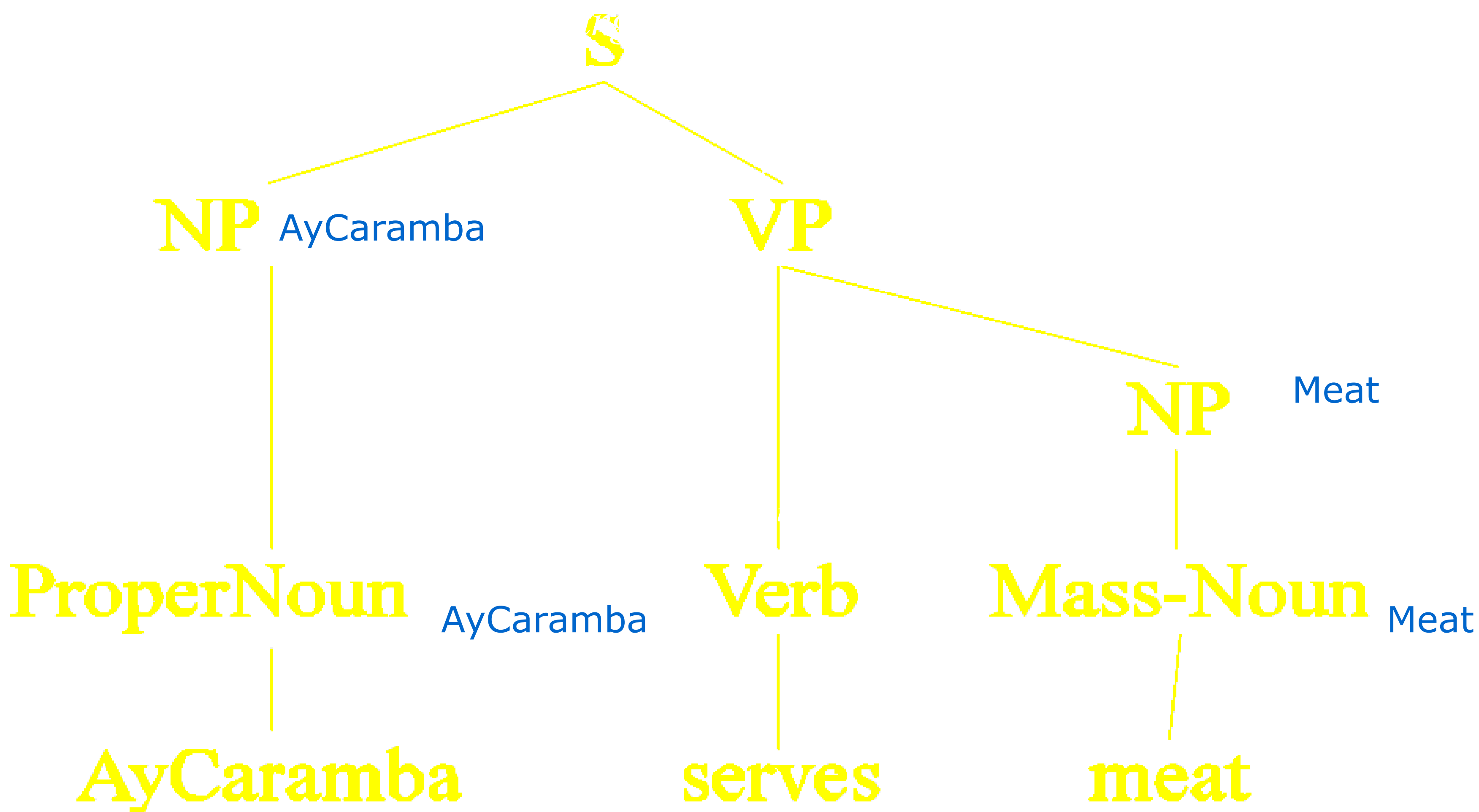
S
NP
VP
NP
ProperNoun
AyCaramba
Verb
Mass-Noun
Meat
AyCaramba
serves
meat

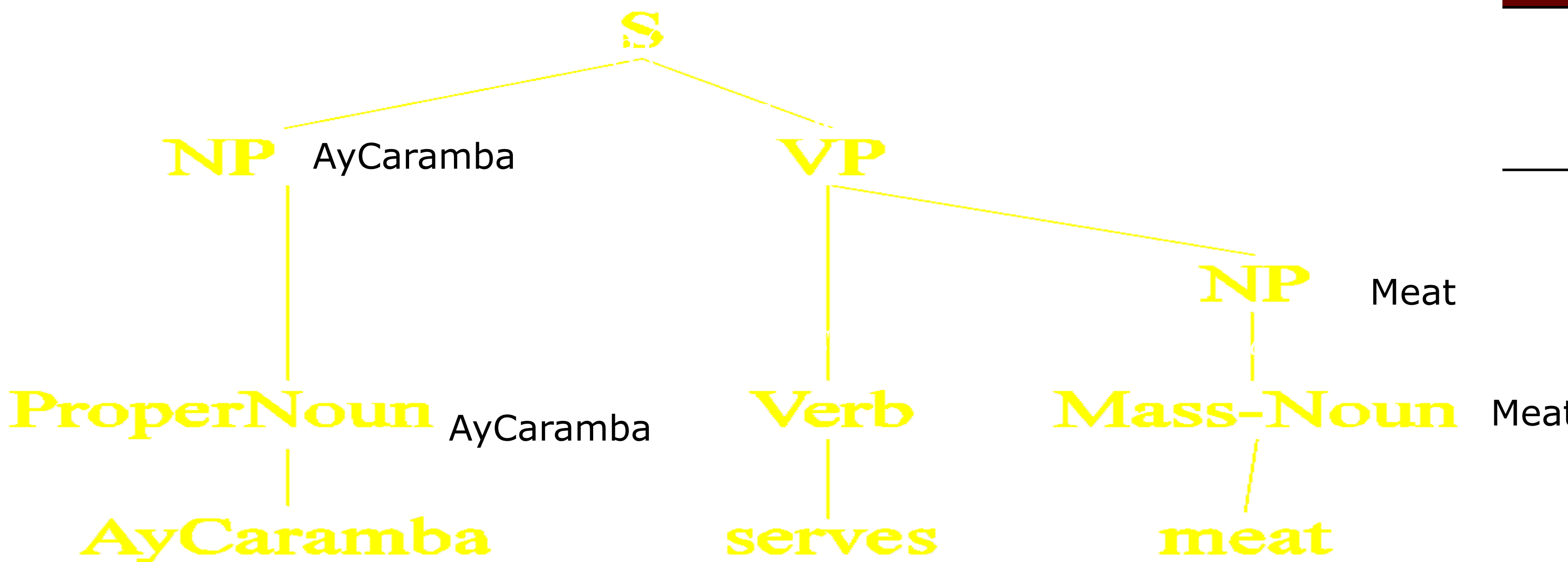
Example



S
NP
AyCaramba
VP
NP
Meat
ProperNoun
AyCaramba
Verb
Mass-Noun
Meat
AyCaramba
serves
meat

Example


S
NP
AyCaramba
VP
ProperNoun
AyCaramba
Verb
NP
Meat
Mass-Noun
Meat
AyCaramba
serves
meat



- **S → NP VP** — **{VP.sem(NP.sem)}**
- **VP → Verb NP** — **{Verb.sem(NP.sem)}**
- **Verb → serves** — $\lambda x\, \lambda y\; \exists e\; Serving(e) \wedge Server(e, y) \wedge Served(e, x)$
- **NP → PropNoun** — **{PropNoun.sem}**
- **NP → MassNoun** — **{MassNoun.sem}**
- **PropNoun → *AyCaramba*** — **{AyCaramba}**
- **MassNoun → *meat*** — **{MEAT}**

Which FOPC representation is better?

$$\lambda x \lambda y \exists e Serving(e) \wedge Server(e,y) \wedge Served(e,x)$$

$$\lambda x \lambda y \exists e Isa(e,Serving) \wedge Server(e,y) \wedge Served(e,x)$$

Possible pop-quiz: Redo previous example using second representation

Thank you

- السلام عليكم ورحمة الله