Slides based on
Jurafsky and Martin
"Speech and Language Processing"

Semantics 2/3
(Semantic Analysis)

Ing. Roberto Tedesco, PhD
roberto.tedesco@polimi.it

NLP – AA 20-21

POLITECNICO MILANO

arcslab
adaptable, relational and cognitive software environments

Semantic analysis: approaches

- Syntax-driven semantic analysis
  - Based on lexicon and grammar
  - Domain independent
- Semantic grammar
  - The elements of the grammar are semantic entities
  - Domain dependent
- Information Extraction
  - Extracts small amount of information
  - Simple approach

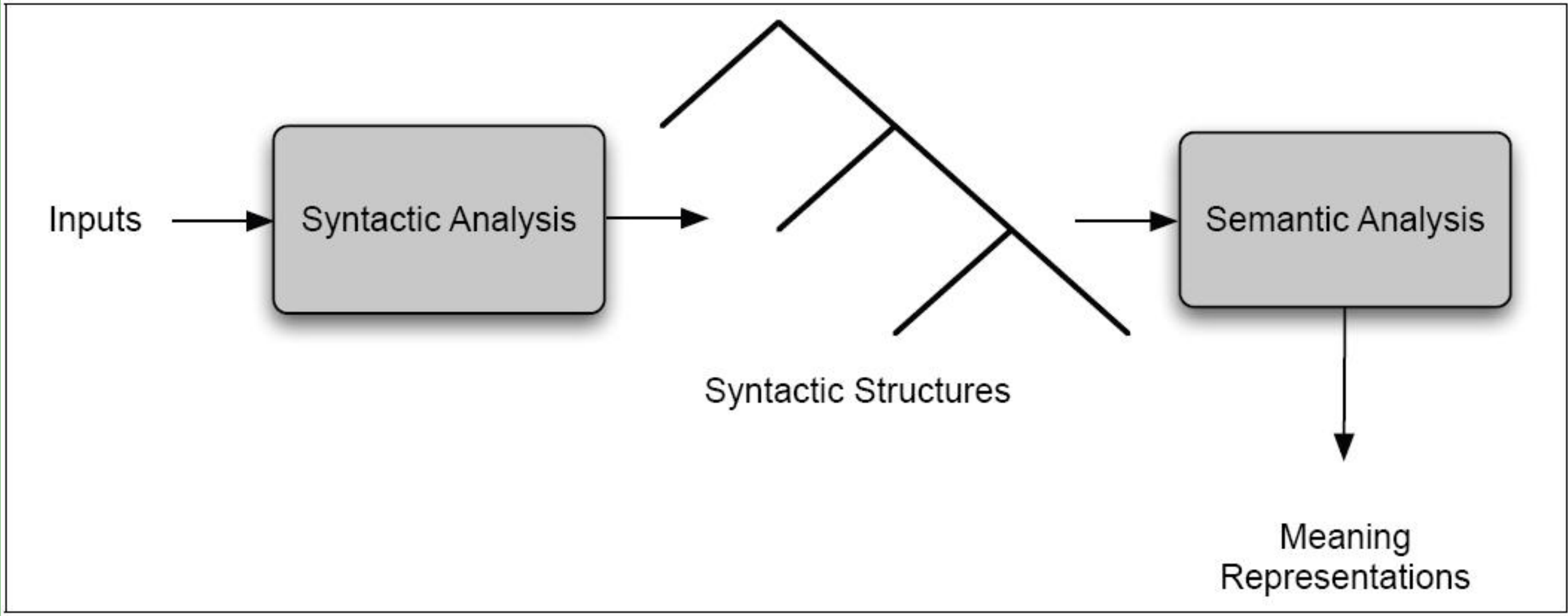
Syntax-based semantic analysis

- Based on the principle of ***compositionality***
  - The meaning of a sentence can be constructed from the meanings of its parts
  - The meaning of a sentence is not based solely on the words that make it up, but also on the ordering and grouping of words, and on the relations among the words in the sentence
- The composition of meaning representations is guided by a grammar:
  - syntactic components
  - relations

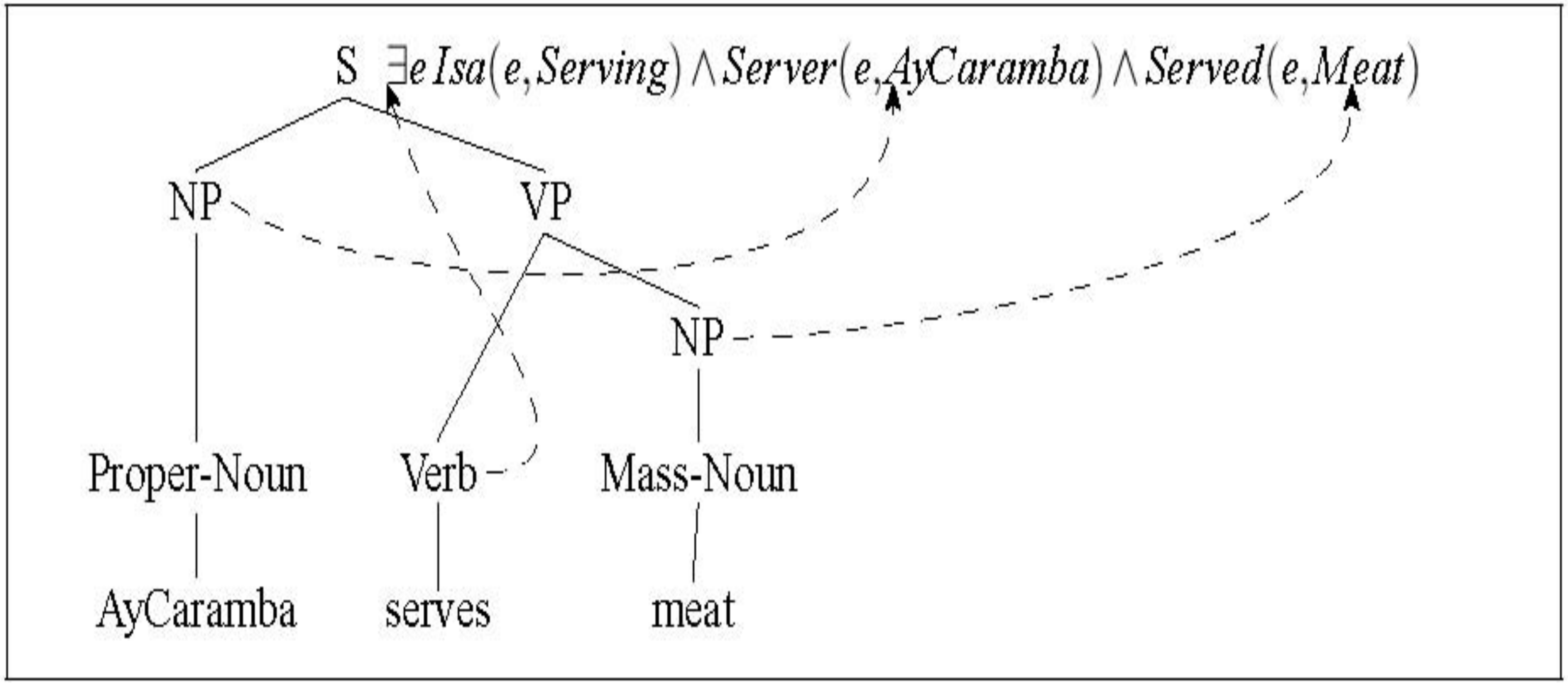
Syntax-based semantic analysis

- Input: parse tree, or feature structures, or lexical dependency diagrams
- Lexical-level ambiguities are not resolved
  - Semantic analysis produces several interpretations
  - Later, lexical semantics analysis will perform WSD
  - In general, a last analysis phase will choose the “right” interpretation

Syntax-based analysis pipeline



Example



- Starting from a parse tree
- If I know that “serves” has roles *Server* and *Served*

$$A.sem = f(\alpha_1.sem, \ldots, \alpha_n.sem)$$

CFG: semantic augmentation

- Augmented CFG rule: $A \rightarrow \alpha_1 \ldots \alpha_n \qquad \{f(\alpha_1 \ldots \alpha_n)\}$

$$ProperNoun \rightarrow AyCaramba \qquad \{\text{AyCaramba}\}$$

$$MassNoun \rightarrow meat \qquad \{\text{Meat}\}$$

$$NP \rightarrow ProperNoun \qquad \{ProperNoun.sem\}$$

$$NP \rightarrow MassNoun \qquad \{MassNoun.sem\}$$

$$Verb \rightarrow serves \qquad \{\lambda x \lambda y\, \exists e\ Isa(e, Serving) \wedge Server(e, y) \wedge Served(e, x)\}$$

$$VP \rightarrow Verb \quad NP \qquad \{Verb.sem(NP.sem)\}$$

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

- The meaning representation assigned to A (called *A.sem*) is computed by function *f* on A's costituents

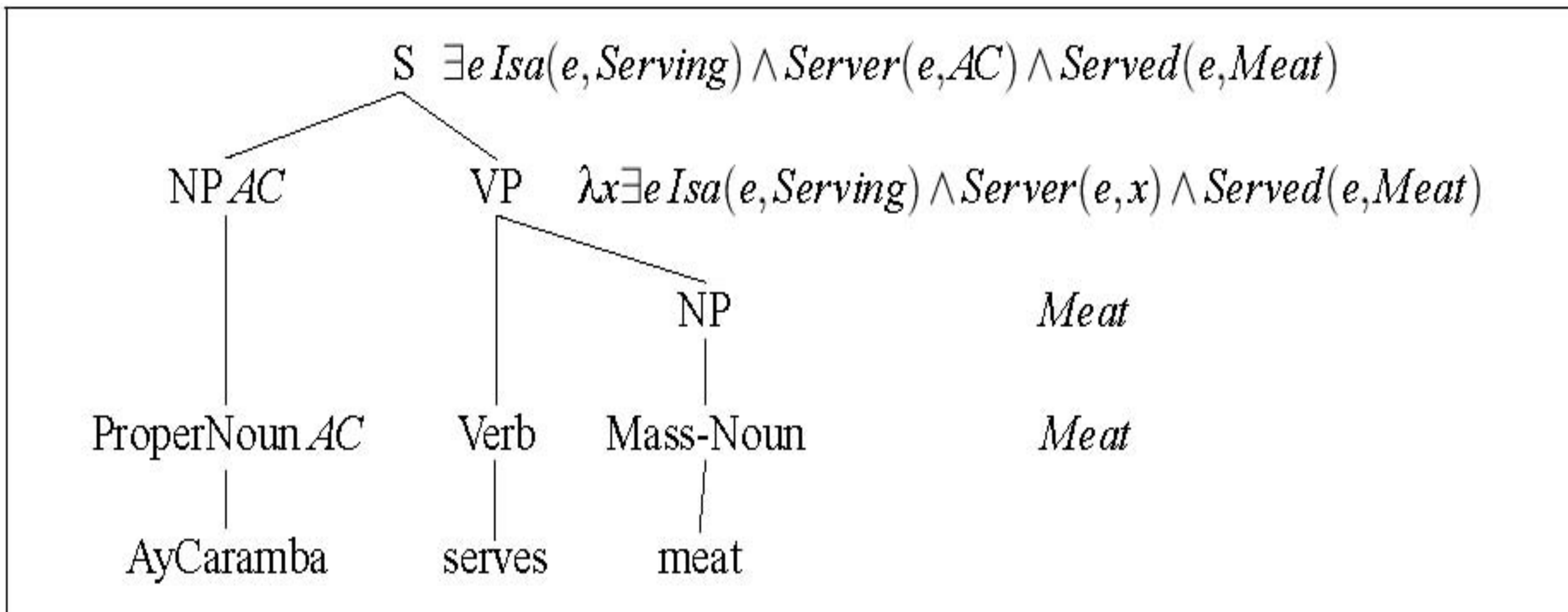
  E.g.: $ProperNoun.sem = AyCaramba.sem = \text{AyCaramba}$

CFG: semantic augmentation

$$\lambda x\, P(x)(A) = P(A)$$

$$\lambda x \lambda y\, P(x,y)(A) = P(A,y)$$

Lambda reduction



Idioms

- The principle of compositionability runs into truble with real language
- Often a sentence is more then the sum of its parts!
- “Coupons are just the tip of the iceberg”
  - The sentence is not speaking of icebergs!
  - Idiom “the tip of the iceberg” → “the beginning”
- Handling idioms: specialized semantic augmentations:

$$NP \rightarrow the\ tip\ of\ the\ iceberg \quad \{Beginning\}$$

Semantic grammars

- Syntactic grammars are not well-suited for the task of semantic analysis
- Mismatch between structures provided by syntactic grammars, and those needed
  1. Key semantic elements are often widely distributed across parse trees [because of issue 2] (this complicates composition)
  2. Parse trees contain syntactically-motivated constituents that play no role in semantics processing (e.g., articles…)
  3. The general nature of syntactic constituents result in semantic attachment that are often too general ("when does it arrives…")
- Semantic grammars permit to overcome these problems
  - Grammars that describe the semantics of sentences

Semantic grammars

"I want to go to eat some Italian food"

- A semantic grammar:

  InfoRequest → *User* want **to go to** eat *FoodType*
  FoodType → *Nationality* *FoodType*

"When does it arrive in Dallas?"

- A semantic grammar:

  InfoRequest → **when does** *Flight* **arrive in** *City*

  - Notice that the "it" has a well-defined semantic meaning
  - Semantic grammars can help with ellipsis and anaphora, because permits to know the expected parts of the sentence

- Main drawback:
  - Semantic grammars are domain-specific, reuse is difficult!

Information Extraction

- Simple approach:
- Search for specific info to extract
- Templates to fill
  - Simple and predefined
  - Can be organized as a hierarchy
- Just a small subset of the text is used, the rest can be discarded
- Common tasks:
  - NER, relation detection, event detection, temporal analysis, template filling

Named entity recognition (NER)

- Anything that can be referred to with a proper name
  - People, organization, location, geo-political entity, facility, vehicle, temporal expression, numerical expression, …

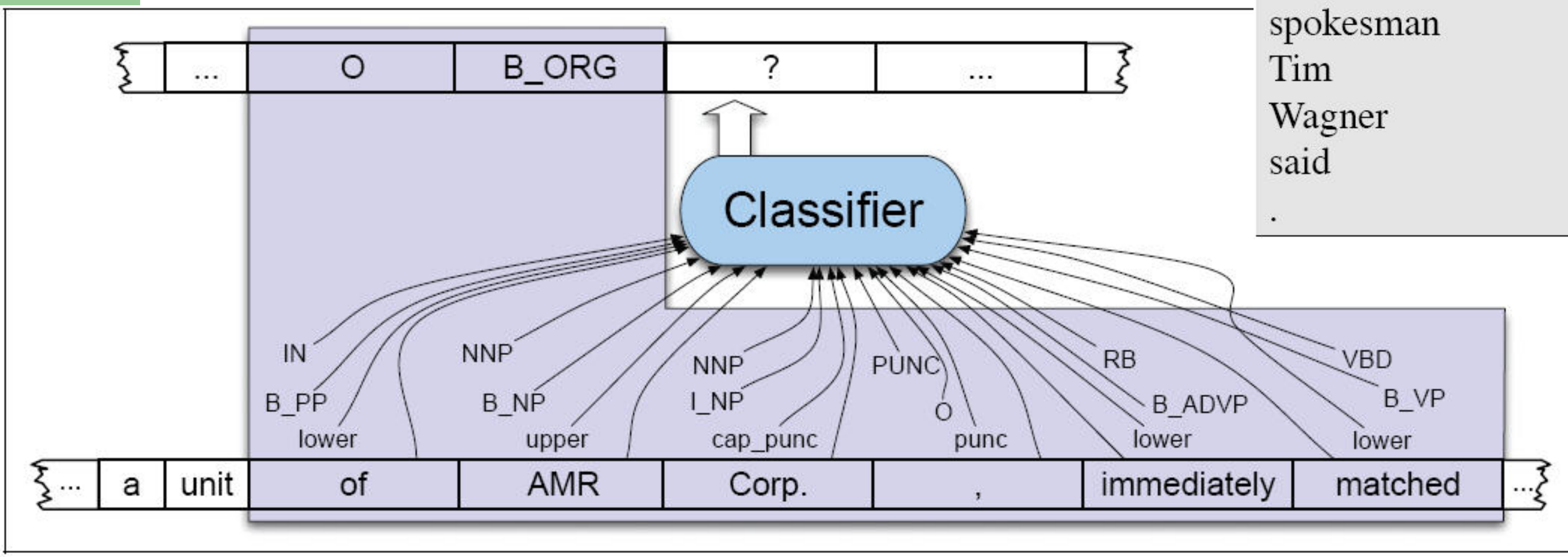
    Citing high fuel prices, [$_{\text{ORG}}$ United Airlines] said [$_{\text{TIME}}$ Friday] it has increased fares by [$_{\text{MONEY}}$ $6]
- Approaches & tools:
  - Regular expressions; e.g.: matching e-mail:

    ([a-z0-9_\.-]+)@([\da-z\.-]+)\.([a-z\.]{2,6})
  - Gazetteers (e.g., list of proper names)
  - Machine learning
- We present a machine learning methodology

NER & machine learning

- Word-by-word sequence labeling
- Features: lexical items, POS, chunk gazetteers, etc.

Words	Label
American	B_{ORG}
Airlines	I_{ORG}
,	O
a	O
unit	O
of	O
AMR	B_{ORG}
Corp.	I_{ORG}
,	O
immediately	O
matched	O
the	O
move	O
,	O
spokesman	O
Tim	B_{PERS}
Wagner	I_{PERS}
said	O
.	O



Relation detection

- Find relations among the entities detected in a text
- Machine learning approach
  - Corpus: texts annotated with relations
  - Important feature: key words, NER, the sentence parse tree

Relations		Examples	Types
Affiliations			
	Personal	*married to, mother of*	PER $\rightarrow$ PER
	Organizational	*spokesman for, president of*	PER $\rightarrow$ ORG
	Artifactual	*owns, invented, produces*	(PER $\mid$ ORG) $\rightarrow$ ART
Geospatial			
	Proximity	*near, on outskirts*	LOC $\rightarrow$ LOC
	Directional	*southeast of*	LOC $\rightarrow$ LOC
Part-Of			
	Organizational	*a unit of, parent of*	ORG $\rightarrow$ ORG
	Political	*annexed, acquired*	GPE $\rightarrow$ GPE

Temporal analysis

- Recognizes:
  - absolute points in time
  - relative times
  - durations
- Approaches:
  - Rule-based systems based on partial parsing or chunking (i.e., searching for patterns)
  - Statistical sequence classifiers based on standard token-by-token IBO encoding
- Normalization: mapping a temporal expression to either a specific point in time, or to a duration
  - "tomorrow at noon" → 2016-1-30T12:00.000+01:00

Event detection

- To identify mentions of events in texts and then assign those events to predefined classes

  [$_{\text{EVENT}}$ Citing] high fuel prices, United Airlines [$_{\text{EVENT}}$ said] Friday it has [$_{\text{EVENT}}$ increased] fares by $6 per round trip

- Most event mentions correspond to verbs
- Approaches:
  - Rule-based and statistical machine learning
  - Both approaches make use of: POS, presence of particular lexical items, and verb tense information

Template filling

- Many texts contain reports of events, and possibly sequences of events, that often correspond to fairly common situations
- These abstract situations can be characterized as *scripts*
- Such scripts can be represented as *templates*
  - Fixed sets of *slots* which take as values *slot-fillers*

FARE-RAISE ATTEMPT:	LEAD AIRLINE:	UNITED AIRLINES
	AMOUNT:	$6
	EFFECTIVE DATE:	2006-10-26
	FOLLOWER:	AMERICAN AIRLINES

Template filling: Machine Learning

- Train separate sequence classifiers for each slot to be filled
- Each classifier recognizes sequences of tokens as potential fillers for its particular slots
- Features: tokens, POS tags, syntactic chunk tags, and named entity tags