Computational Semantics 2

CS224N

Christopher Manning

(Borrows some slides from Mary Dalrymple, Jason Eisner, and Jim Martin)

Central Problems

Parsing

Learning

Modeling

Compositional Semantics

- We've discussed what semantic representations should look like.
- **But how do we get them from sentences???**
- First - parse to get a syntax tree.
- Second - look up the semantics for each word.
- Third - build the semantics for each constituent
  - Work from the bottom up
  - The syntax tree is a "recipe" for how to do it
- Principle of Compositionality
  - The meaning of a whole is derived from the meanings of the parts, via composition rules

A simple grammar of English

(in Definite Clause Grammar, DCG, form – as in Prolog)

sentence --> noun_phrase, verb_phrase.

noun_phrase --> proper_noun.

noun_phrase --> determiner, noun.

verb_phrase --> verb, noun_phrase.

proper_noun --> [John]

proper_noun --> [Mary]

determiner --> [the]

determiner--> [a]

verb --> [ate]

verb --> [kissed]

noun --> [cake]

noun --> [lion]

Extending the grammar to check number agreement between subjects and verbs

```
S --> NP(Num), VP(Num).
NP(Num) --> proper_noun(Num).
NP(Num) --> det(Num), noun(Num).
VP(Num) --> verb(Num), NP(_).
```

```
proper_noun(s) --> [Mary].
det(s) --> [the].
det(p) --> [the].
```

```
noun(s) --> [lion].
noun(p) --> [lions].
verb(s) --> [eats].
verb(p) --> [eat].
```

A simple DCG grammar with semantics

sentence(SMeaning) --> noun_phrase(NPMeaning), verb_phrase(VPMeaning), {combine (NPMeaning, VPMeaning, SMeaning)}.

verb_phrase(VPMeaning) --> verb(Vmeaning), noun_phrase(NPMeaning), {combine (NPMeaning, VMeaning, VPMeaning)}.

noun_phrase (NPMeaning) --> name(NPMeaning).

name(john) --> [john].	verb(λx.jumps(x)) --> [jumps]
name(mary) --> [mary].	verb(λy.λx.loves(x,y)) -->[loves]

Combine(X, Y, Z) --> apply(Y, X, Z)

Formal Compositional Semantics …

- **Richard Montague** (1930-1971)

- *"… I reject the contention that an important theoretical difference exists between formal and natural languages …"*

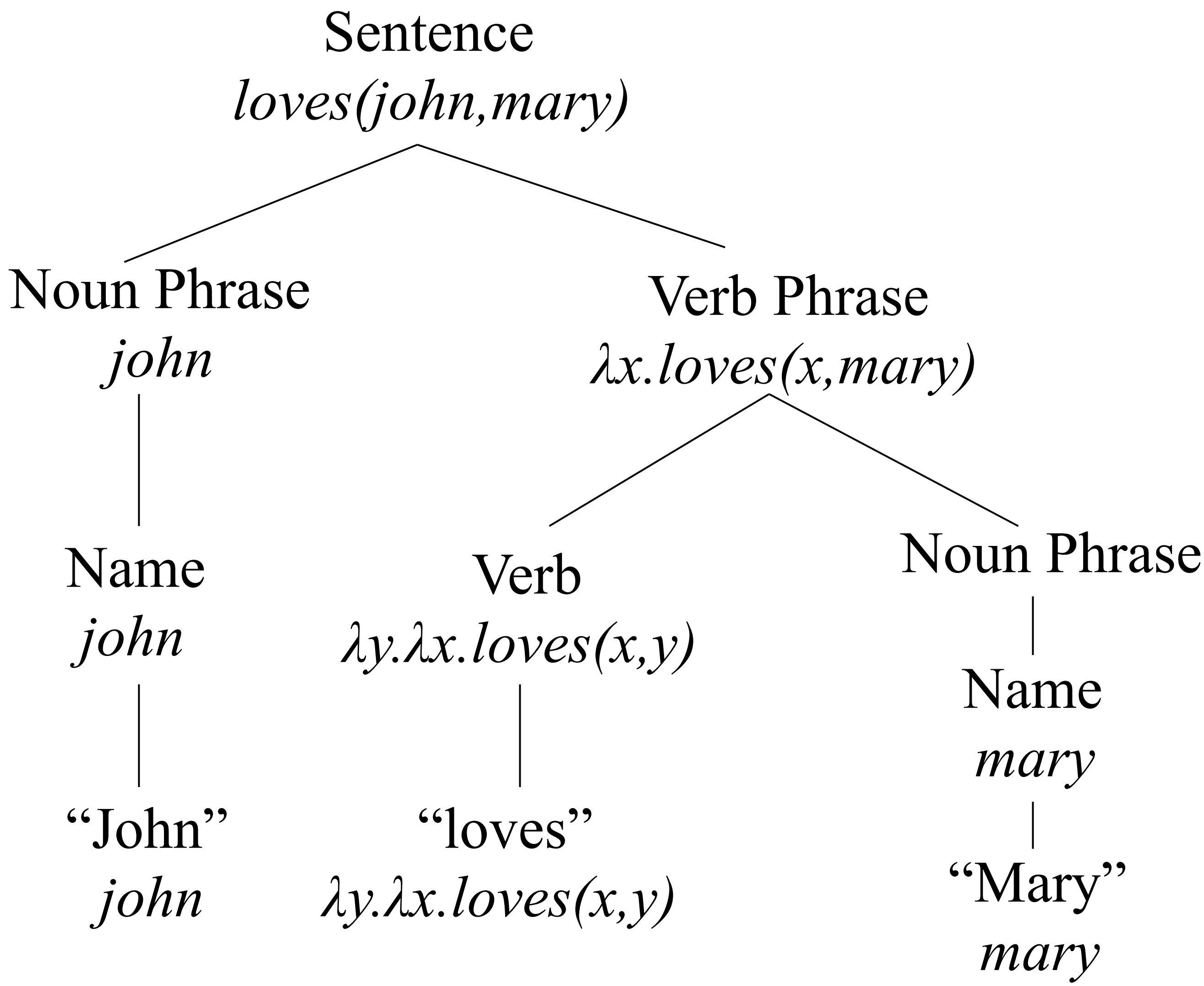
Augmented CFG Rules

- We can also accomplish this just by attaching semantic formation rules to our syntactic CFG rules

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

- 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.
- The functions/operations permitted in the semantic rules are restricted, falling into two classes
  - Pass the semantics of a daughter up unchanged to the mother
  - Apply (as a function) the semantics of one of the daughters of a node to the semantics of the other daughters

Parse tree with associated semantics



In detail: Beta-Reduction

$(\lambda y \lambda x.\text{love}(x,y)[\text{mary}][\text{john}]$

$\beta\Rightarrow (\lambda x.\text{love}(x,\text{mary}))[\text{john}]$

$\beta\Rightarrow \text{love}(\text{john},\text{mary})$

How do things get more complex? (The former) GRE analytic section

- Six sculptures – C, D, E, F, G, H – are to be exhibited in rooms 1, 2, and 3 of an art gallery.
  - Sculptures C and E may not be exhibited in the same room.
  - Sculptures D and G must be exhibited in the same room.
  - If sculptures E and F are exhibited in the same room, no other sculpture may be exhibited in that room.
  - At least one sculpture must be exhibited in each room, and no more than three sculptures may be exhibited in any room.
- If sculpture D is exhibited in room 3 and sculptures E and F are exhibited in room 1, which of the following may be true?
  1. Sculpture C is exhibited in room 1.
  2. Sculpture H is exhibited in room 1.
  3. Sculpture G is exhibited in room 2.
  4. Sculptures C and H are exhibited in the same room.
  5. Sculptures G and F are exhibited in the same room.

Scope Needs to be Resolved!

At least one sculpture must be exhibited in each room.

The same sculpture in each room?

No more than three sculptures may be exhibited in any room.

Reading 1: For every room, there are no more than three sculptures exhibited in it.

Reading 2: Only three or less sculptures are exhibited (the rest are not shown).

Reading 3: Only a certain set of three or less sculptures may be exhibited in any room (for the other sculptures there are restrictions in allowable rooms).

- Some readings will be ruled out by being uninformative or by contradicting other statements
- Otherwise we must be content with distributions over scope-resolved semantic forms

Generalized Quantifiers

- Generalized quantifiers are a relation between two properties (predicates on entities)
- most(pig, big) = "most pigs are big"
    - Equivalently, most(λx pig(x), λx big(x))
  - returns true if most of the things satisfying the first predicate also satisfy the second predicate
- similarly for other quantifiers
  - all(pig,big) (equivalent to $\forall x\ pig(x) \Rightarrow big(x)$)
    - the set of pigs are a subset of the set of big things
  - a(pig,big) (equivalent to $\exists x\ pig(x)$ AND $big(x)$)
  - can even build complex quantifiers from English phrases:
    - "between 12 and 75"; "a majority of"; "all but the smallest 2"

An alternative: Semantic Grammars

- A problem with traditional linguistic grammars is that they don't necessarily reflect the semantics in a straightforward way
- You can deal with this by…
  - Fighting with the grammar
    - Complex lambdas and complex terms, etc.
  - Rewriting the grammar to reflect the semantics
    - And in the process give up on some syntactic niceties
    - known as "Semantic grammars"
      - Simple idea, dumb name

Lifer Semantic Grammars

- Example domain—access to DB of US Navy ships
  S → <present> the <attribute> of <ship>
  <present> → what is | [can you] tell me
  <attribute> → length | beam | class
  <ship> → the <shipname>
  <shipname> → kennedy | enterprise
  <ship> → <classname> class ships
  <classname> →kitty hawk | lafayette
- Example inputs recognized by above grammar:
  can you tell me the class of the Enterprise
  what is the length of Kitty Hawk class ships
  - Many categories are not "true" syntactic categories
  - Words are recognized by their context rather than category (e.g. *class*)
  - Recognition is strongly directed
  - Strong direction useful for error detection and correction
    - G. Hendrix, E. Sacerdoti, D. Sagalowicz, and J.Slocum. 1978. Developing a natural language interface to complex data. ACM Transactions on Database Systems 3:105-147

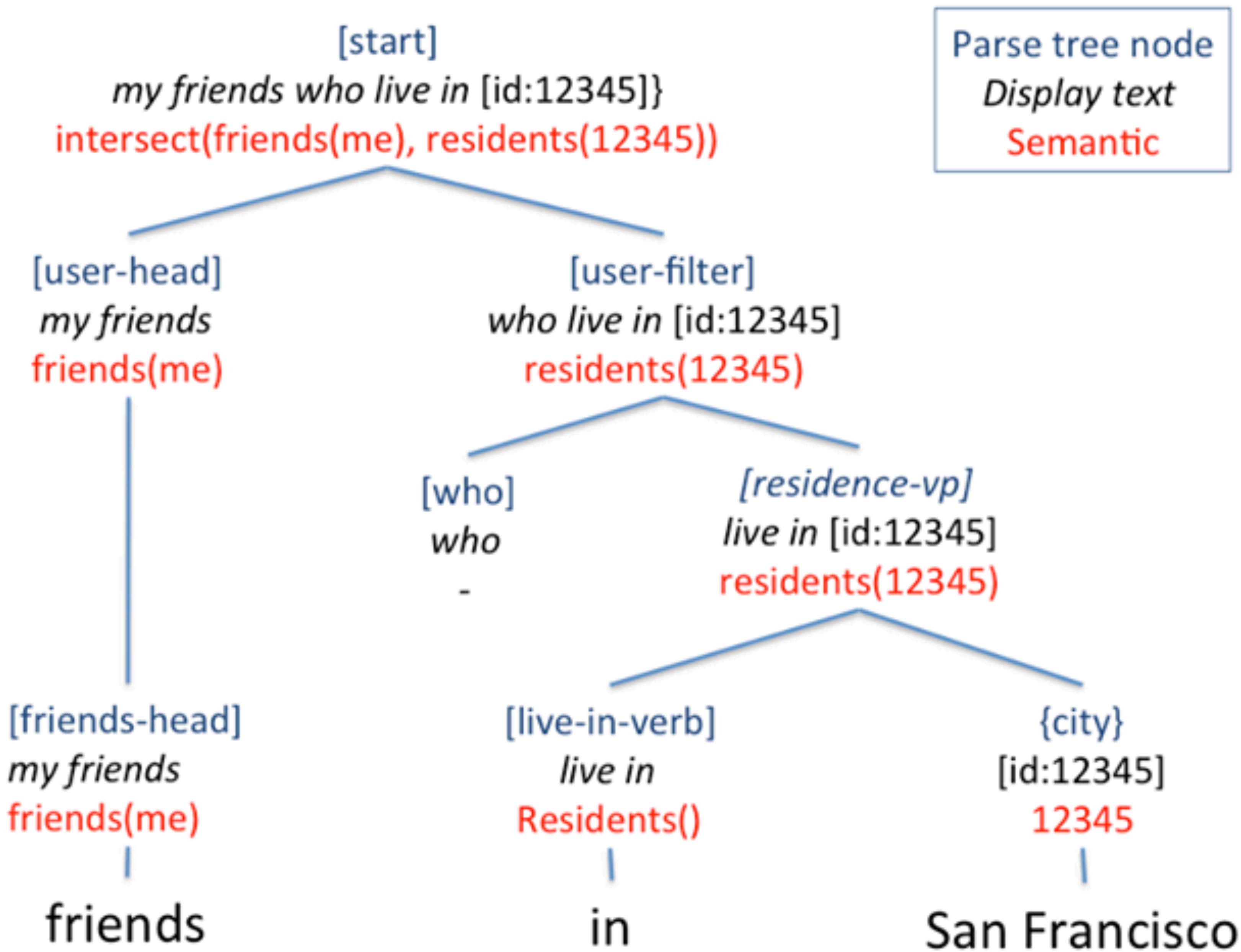
Semantic Grammar

- The term semantic grammar refers to the motivation for the grammar rules
  - The technology (plain CFG rules with a set of terminals) is the same as we’ve been using
  - The good thing about them is that you get exactly the semantic rules you need
  - The bad thing is that you need to develop a new grammar for each new domain
- Typically used in conversational agents in constrained domains
  - Limited vocabulary
  - Limited grammatical complexity
  - Syntactic parsing can often produce all that’s needed for semantic interpretation even in the face of “ungrammatical” input – write fragment rules

Facebook Graph Search

- Uses a weighted context free grammar (WCFG) to represent the Graph Search query language:
- [start] => [users] $1
- [users] => my friend friends(me)
- [users] => friends of [users] friends($1)
- [users] => {user} $1
- [start] => [photos] $1
- [photos] => photos of [users] photos($1)
- A terminal symbol can be an entity, e.g., {user}, {city}, {employer}, {group}; it can also be a word/phrase, e.g., friends, live in, work at, members, etc. A parse tree is produced by starting from [start] and expanding the production rules until it reaches terminal symbols.

https://www.facebook.com/notes/facebook-engineering/under-the-hood-the-natural-language-interface-of-graph-search/10151432733048920
http://spectrum.ieee.org/telecom/internet/the-making-of-facebooks-graph-search



The parse tree, semantic and entity ID used in the above example are for illustration only; they do not represent real information used in Graph Search Beta

Semantic Grammars Summary

- Advantages:
  - Efficient recognition of limited domain input
  - Absence of overall grammar allows pattern-matching possibilities for idioms, etc.
  - No separate interpretation phase
  - Strength of top-down constraints allows powerful ellipsis mechanisms

    What is the length of the Kennedy? The Kittyhawk?
- Disadvantages:
  - Different grammar required for each new domain
  - Lack of overall syntax can lead to "spotty" grammar coverage
    - E.g. fronting possessive in "<attribute> of <ship>" to <ship> 's <attribute> doesn't imply fronting in "<rank> of <officer>"
  - Difficult to develop grammars past a certain size
  - Suffers from fragility