School of Computing

Semantics

CS4248 Natural Language Processing

Week 10

Shreyas KUTHANOOR PRAKASH and Min-Yen KAN

10

Recap of Week 09

Context Free Grammars
Conversion to Chomsky Normal Form (CNF)

Parsing – from 1D sequence to 2D trees

Syntactic Parsing via CKY
Probabilistic Syntactic Parsing

NUS National University of Singapore | School of Computing

Week 10 Agenda

Introduction to Semantics
Computational Lexical Semantics
WordNet 3.0
Word Similarity, Redux

Meaning Representation
First Order Logic
Lambda Calculus
Adjuncts and Roles

Introduction to Semantics

School of Computing

Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

Semantic Representation

Meaning Representation: Formal structures composed from symbols to represent the meaning of sentences

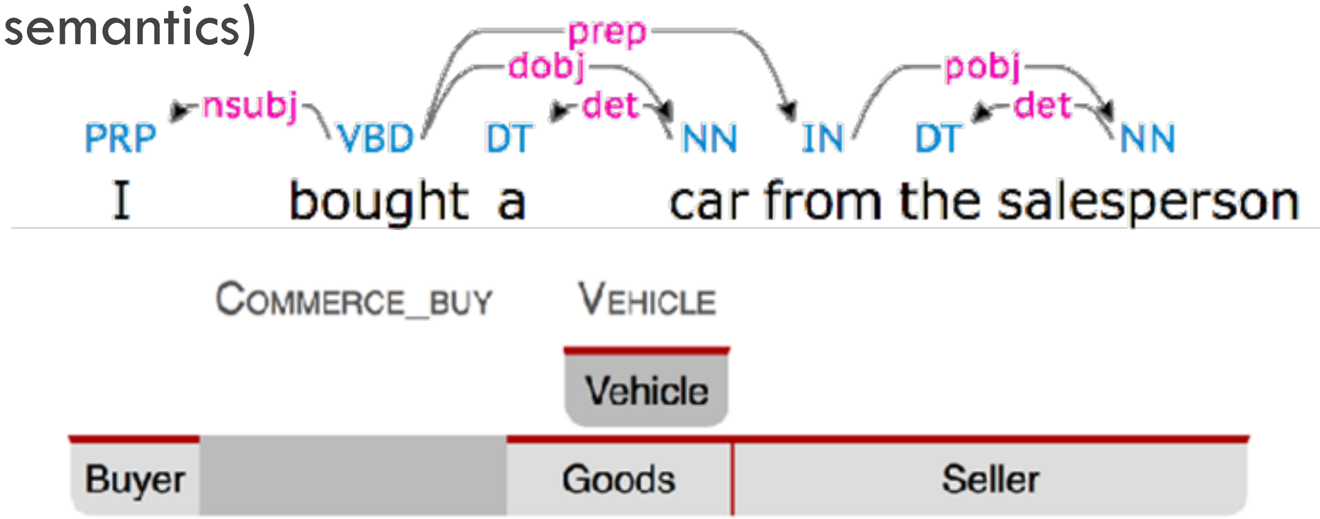
We will be dealing with two types of semantics:

- **Lexical semantics:** representing the meaning of words (and their relations)
- **Logical semantics:** representing the meaning of sentences

Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

Importance of syntax

Syntax serves as the foundation for semantic analysis (on many levels of representation: semantic roles, compositional semantics, frame semantics)

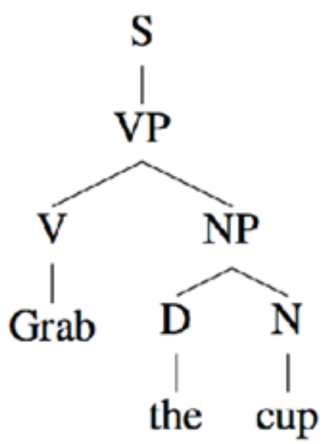


Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

Is syntax sufficient?

Syntax encodes the structure of language, but doesn't directly address meaning.

For example, let's look at the sentence: *Grab the cup.*



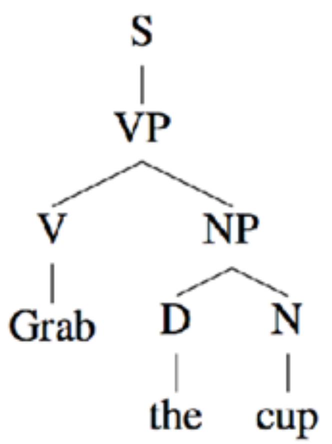
Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

Is syntax sufficient?

For example, let's look at the sentence: *Grab the cup.*

Syntax alone doesn't ground *Grab* in an action to take in the world.

Grounding: establishing the time, location or actuality of a situation according to some reference point.

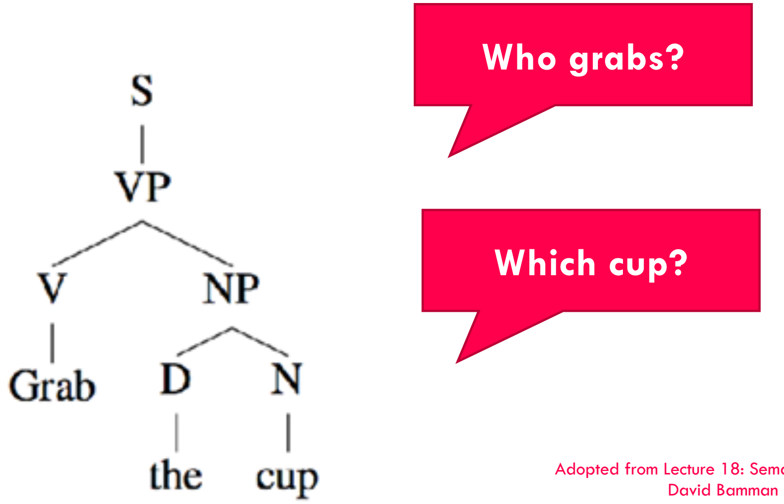


Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

Is syntax sufficient?

For example, let's look at the sentence: *Grab the cup.*

Syntax alone doesn't ground *Grab* in an action to take in the world.



Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

Lexical semantics

Vector representation that encodes information about the distribution of contexts a word appears in.

Words that appear in similar contexts have similar representations (and similar meanings, by the distributional hypothesis).

We can represent what individual words "mean" as a function of what other words they're related to (but that's still not grounding).

Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

Nearest embeddings

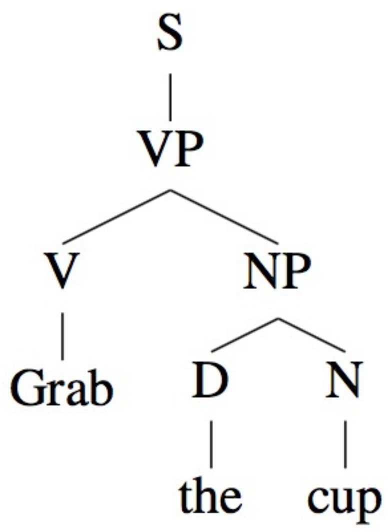
grab	1
throw	0.824
pull	0.818
knock	0.799
grabbing	0.789
steal	0.787
pulling	0.764
grabs	0.756
away	0.746
catch	0.74

Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

Why is syntax insufficient?

Even if we have a reference model for each word in a sentence, syntax doesn't tell us how those referents changes as a function of their compositionally.

e.g. What is the meaning of a VP?



Computational Lexical Semantics

WordNet and other ontologies

Adapted from Dan Jurafsky (Stanford)

Computational Lexical Semantics

Any computational process involving word meaning!

- Computing Word Similarity
  - Distributional (Vector) Models of Meaning
- Computing Word Relations
- Word Sense Disambiguation
- Semantic Role Labeling
- Computing word connotation and sentiment

Adapted from Dan Jurafsky (Stanford)

Synonyms and near-synonymy: computing the similarity between words

"*fast*" is similar to "*rapid*"

"*tall*" is similar to "*height*"

Used in applications such as Question Answering:

Q: "How tall is Mt. Everest?"
Candidate A: "The official height of Mount Everest is 8848 meters"

Adapted from Dan Jurafsky (Stanford)

Word Relations:
Part–Whole or Supertype–Subtype

A "*collie*" is-a "*dog*"

A "*wheel*" is-part-of a "*car*"

Uses:

- Question Answering:
  Q: Does Sean have a dog? Candidate A: "Sean has two collies"
- Reference resolution
  Speaker A: "How's your car?"
  Speaker B: "I'm having problems with the wheels"

Adapted from Dan Jurafsky (Stanford)

Lemmas have senses

NUS National University of Singapore | School of Computing

One lemma "*bank*" can have many meanings:

Sense 1: …a **bank**$_1$ can hold the investments in a custodial account…

Sense 2: …as agriculture burgeons on the east **bank**$_2$ the river will shrink even more

Sense (or word sense)

- A discrete representation of an aspect of a word's meaning.
- The lemma *bank* here has two senses.

Adapted from Dan Jurafsky (Stanford)

Homonymy

Words that share a form but have unrelated, distinct meanings:

- *bank*$_1$: financial institution, *bank*$_2$: sloping land
- *bat*$_1$: club for hitting a ball, *bat*$_2$: nocturnal flying mammal

1. Homographs (bank/bank, bat/bat)
2. Homophones:
   1. *Write* and *right*
   2. *Piece* and *peace*

Adapted from Dan Jurafsky (Stanford)

Homonymy behaving badly

c.f. Week 1's *Expressivity*

Information retrieval

- *bat care*

Machine Translation

- bat: *murciélago* (animal) or *bate* (for baseball)

Text-to-Speech

- *bass* (stringed instrument) vs. *bass* (fish)

Adapted from Dan Jurafsky (Stanford)

Metonymy:
A systematic relationship between senses

Lots of types of polysemy are systematic

- *School*, *University*, *Hospital*
- All can mean the institution or the building.

A systematic relationship:

- Building ⟺ Organization

Other such kinds of systematic polysemy:

- Author (Austen wrote “Emma”) ⟺ Works of Author (I love Austen)
- Tree (Plums have pretty blossoms) ⟺ Fruit (I ate a preserved plum)

Adapted from Dan Jurafsky (Stanford)

How do we know when a word has more than one sense?

The zeugma test: Two senses of *serve*?

- *Which flights serve breakfast?*
- *Does Lufthansa serve Philadelphia?*
- *Does Lufthansa serve breakfast and San Jose?*

Since this conjunction sounds weird, we say that these are two different senses of *serve*

Synonyms

Word that have the same meaning in some or all contexts.

- *filbert / hazelnut*
- *couch / sofa*
- *big / large*
- *automobile / car*
- *vomit / throw up*
- *water / $H_2 0$*

What are some cases of these exceptions here?

But there are few examples of perfect synonymy.

- Even if many aspects of meaning are identical.
- Still may not preserve acceptability based on notions of politeness, slang, register, genre, etc.

Adapted from Dan Jurafsky (Stanford)

Synonymy is a relation between senses rather than words

Consider the words *big* and *large* 🤔

Are they synonyms?

- *How big is that plane?*
- *Would I be flying on a large or small plane?*

How about here?

- *Miss Nelson became a kind of big sister to Benjamin.*
- *Miss Nelson became a kind of large sister to Benjamin.*

Why?

- *big* has a sense that means being older, or grown up.
- *large* lacks this sense.

Adapted from Dan Jurafsky (Stanford)

Hyponymy and Hypernymy

- One sense is a hyponym of another if the first sense is more specific, denoting a subclass of the other
  - *car* is a hyponym of *vehicle*
  - *mango* is a hyponym of *fruit*
- Conversely hypernym/superordinate ("hyper is super")
  - *vehicle* is a hypernym of *car*
  - *fruit* is a hypernym of *mango*

Superordinate/hyper	*vehicle*	*fruit*	*furniture*
Subordinate/hyponym	*car*	*mango*	*chair*

Adapted from Dan Jurafsky (Stanford)

Meronymy

The part–whole relation

- A *leg* is part of a *chair*; a *wheel* is part of a *car*.
- *wheel* is a **meronym** of *car*, and *car* is a **holonym** of *wheel*.

Adapted from Dan Jurafsky (Stanford)

Hyponymy more formally

Extensional:

- The class denoted by the superordinate extensionally includes the class denoted by the hyponym

Entailment:

- A sense A is a hyponym of sense B if *being an* A entails *being a* B

Hyponymy is usually transitive

- (A hypo B and B hypo C entails A hypo C)

- Another name: the IS-A hierarchy
  - A IS-A B (or A ISA B)
  - B **subsumes** A

Adapted from Dan Jurafsky (Stanford)

Hyponyms and Instances

- WordNet has both classes and instances.
- An **instance** is an individual, a proper noun that is a unique entity
    - `San Francisco` is an **instance** of `city`
  - But `city` is a class
    - `city` is a **hyponym** of `municipality...location...`

Adapted from Dan Jurafsky (Stanford)

WordNet 3.0

http://wordnetweb.princeton.edu

Adapted from Dan Jurafsky (Stanford)

WordNet 3.0

NUS National University of Singapore | School of Computing

A hierarchically organized lexical database

On-line thesaurus + aspects of a dictionary

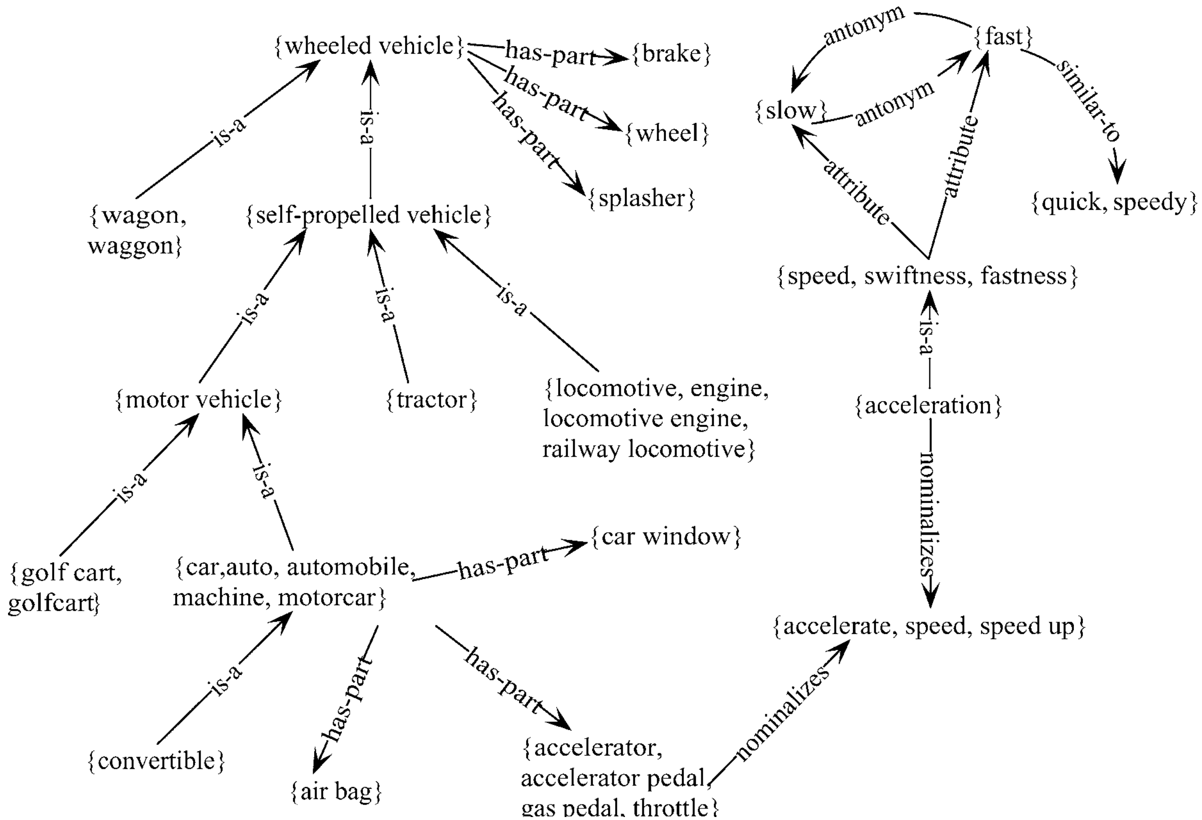
- Other languages also available or under development

Category	Unique Strings
Noun	117,798
Verb	11,529
Adjective	22,479
Adverb	4,481

Adapted from Dan Jurafsky (Stanford)

WordNet: Online thesaurus

Many of the lexical relationships we discussed are compiled in WN3



Adapted from Dan Jurafsky (Stanford)

Senses in WordNet

The synset (synonym set), the set of near-synonyms, instantiates a sense or concept, with a gloss

- E.g., *chump* (noun): "a person who is gullible and easy to take advantage of"

This sense of "*chump*" is shared by 9 words:

- *chump*$_1$, *fool*$_2$, *gull*$_1$, *mark*$_9$, *patsy*$_1$, *fall guy*$_1$, *sucker*$_1$, *soft touch*$_1$, *mug*$_2$

Each of these senses have this same gloss

- Not every sense; Sense 2 of *gull* (*gull*$_2$) is the aquatic bird.

Hypernym Hierarchy for *bass*

- S: (n) **bass**, basso (an adult male singer with the lowest voice)
  - *direct hypernym* | ***inherited hypernym*** | *sister term*
    - S: (n) singer, vocalist, vocalizer, vocaliser (a person who sings)
      - S: (n) musician, instrumentalist, player (someone who plays a musical instrument (as a profession))
        - S: (n) performer, performing artist (an entertainer who performs a dramatic or musical work for an audience)
          - S: (n) entertainer (a person who tries to please or amuse)
            - S: (n) person, individual, someone, somebody, mortal, soul (a human being) *"there was too much for one person to do"*
              - S: (n) organism, being (a living thing that has (or can develop) the ability to act or function independently)
                - S: (n) living thing, animate thing (a living (or once living) entity)
                  - S: (n) whole, unit (an assemblage of parts that is regarded as a single entity) *"how big is that part compared to the whole?"; "the team is a unit"*
                    - S: (n) object, physical object (a tangible and visible entity; an entity that can cast a shadow) *"it was full of rackets, balls and other objects"*
                      - S: (n) physical entity (an entity that has physical existence)
                        - S: (n) entity (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

WordNet Noun Relations

Relation	Also Called	Definition	Example
Hypernym	Superordinate	From concepts to superordinates	$breakfast^1 \rightarrow meal^1$
Hyponym	Subordinate	From concepts to subtypes	$meal^1 \rightarrow lunch^1$
Instance Hypernym	Instance	From instances to their concepts	$Austen^1 \rightarrow author^1$
Instance Hyponym	Has-Instance	From concepts to concept instances	$composer^1 \rightarrow Bach^1$
Member Meronym	Has-Member	From groups to their members	$faculty^2 \rightarrow professor^1$
Member Holonym	Member-Of	From members to their groups	$copilot^1 \rightarrow crew^1$
Part Meronym	Has-Part	From wholes to parts	$table^2 \rightarrow leg^3$
Part Holonym	Part-Of	From parts to wholes	$course^7 \rightarrow meal^1$
Substance Meronym		From substances to their subparts	$water^1 \rightarrow oxygen^1$
Substance Holonym		From parts of substances to wholes	$gin^1 \rightarrow martini^1$
Antonym		Semantic opposition between lemmas	$leader^1 \Longleftrightarrow follower^1$
Derivationally Related Form		Lemmas w/same morphological root	$destruction^1 \Longleftrightarrow destroy^1$

Adapted from Dan Jurafsky (Stanford)

WordNet Verb Relations

Relation	Definition	Example
Hypernym	From events to superordinate events	*fly*^{9} → *travel*^{5}
Troponym	From events to subordinate event (often via specific manner)	*walk*^{1} → *stroll*^{1}
Entails	From verbs (events) to the verbs (events) they entail	*snore*^{1} → *sleep*^{1}
Antonym	Semantic opposition between lemmas	*increase*^{1} ⟺ *decrease*^{1}
Derivationally Related Form	Lemmas with same morphological root	*destroy*^{1} ⟺ *destruction*^{1}

Adapted from Dan Jurafsky (Stanford)

Word Similarity, Redux

By the book (thesaurus, actually)
via WordNet::Similarity and NLTK

Adapted from Dan Jurafsky (Stanford)

Two classes of similarity algorithms

Distributional algorithms

- Do words have similar distributional contexts?

Thesaurus-based algorithms

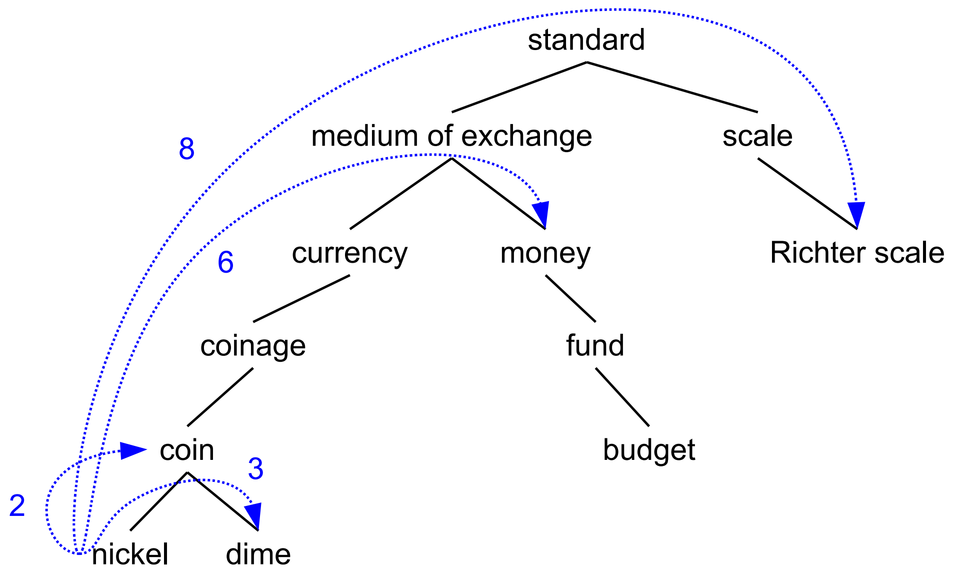
- Are words “nearby” in a hypernym hierarchy?
- Do words have similar glosses (definitions)?

Adapted from Dan Jurafsky (Stanford)

Path based similarity

Two concepts (senses/synsets) are similar if they are near each other in the thesaurus hierarchy.

- Have a short path between them.
- Concepts have path of 1 to themselves.



Adapted from Dan Jurafsky (Stanford)

Example: path-based similarity

$$simPath(c_1, c_2) = 1 \,/\, pathlength(c_1, c_2)$$

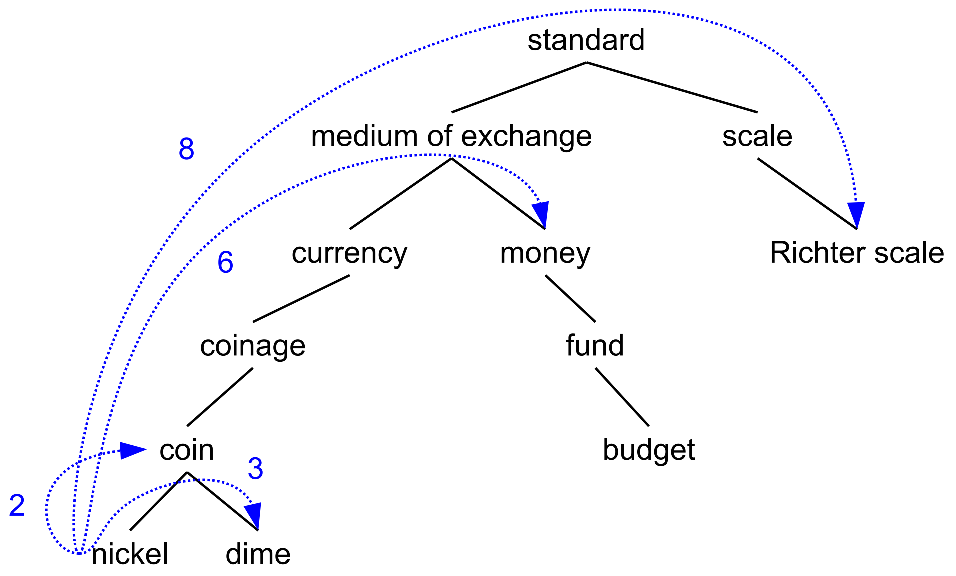
$simPath$(nickel,coin)
= 1/2 = 0.5

$simPath$(fund,budget)
= 1/2 = 0.5

$simPath$(nickel,currency)
= 1/4 = 0.25

$simPath$(nickel,money)
= 1/6 = 0.17

$simPath$(coinage, Richter scale)
= 1/6 = 0.17



Adapted from Dan Jurafsky (Stanford)

Problem with basic path similarity

Assumes each link represents a uniform distance

- But *nickel* to *money* seems to us to be closer than *nickel* to *standard*
- Nodes high in the hierarchy are very abstract

We instead want a metric that

- Represents the cost of each edge independently
- Words connected only through abstract nodes are less similar

Adapted from Dan Jurafsky (Stanford)

Entropy to the rescue
Information Content (Resnik, '95)

Let's define $P(c)$ as the probability that a randomly selected word in a corpus is an instance of concept c

Formally: there is a distinct random variable, ranging over words, associated with each concept in the hierarchy

Then, for a given concept, each observed noun is either
- a member of that concept with probability $P(c)$
- not a member of that concept with probability $1 - P(c)$

All words are members of the root node (Entity)
- $P(root) = 1$

The lower a node in hierarchy, the lower its probability

Adapted from Dan Jurafsky (Stanford)

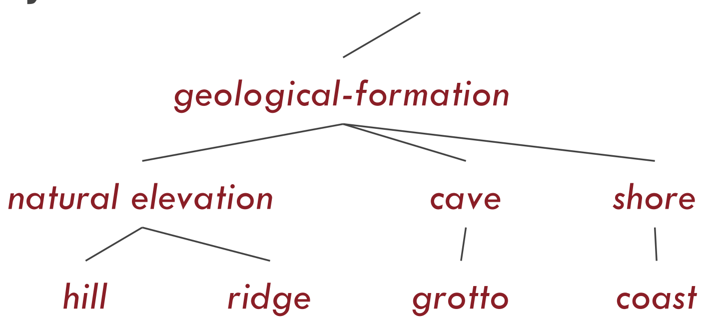
$P(c)$ from corpora

Each instance of *hill₁* counts toward frequency of natural elevation, geological formation, entity, etc.

Let $words(c)$ be the set of all words that are children of node c

- words("*geo-formation*") = { *hill*, *ridge*, *grotto*, *coast*, … }
- words("*natural elevation*") = { *hill*, *ridge* }

$$P(c) = \frac{\sum_{w \in words(c)} count(w)}{N}$$



Adapted from Dan Jurafsky (Stanford)

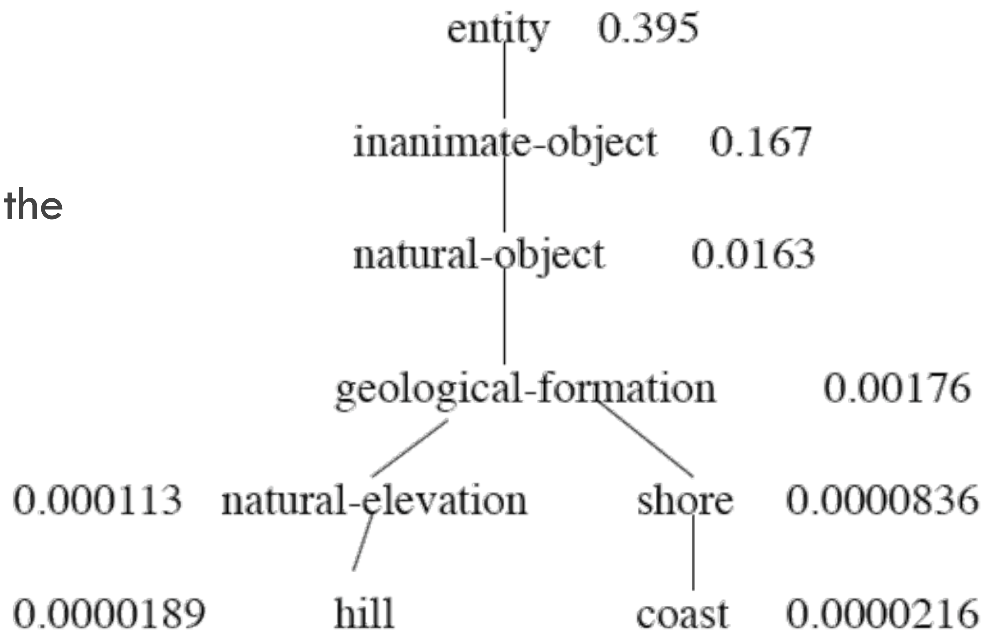
WordNet hierarchy with $P(c)$

Information content

- $IC(c) = -\log P(c)$

Lowest common subsumer

- $LCS(c_1, c_2) =$ The most informative (lowest) node in the hierarchy subsuming both c_1 and c_2.



Adapted from Dan Jurafsky (Stanford)

WordNet hierarchy with $P(c)$

Information content

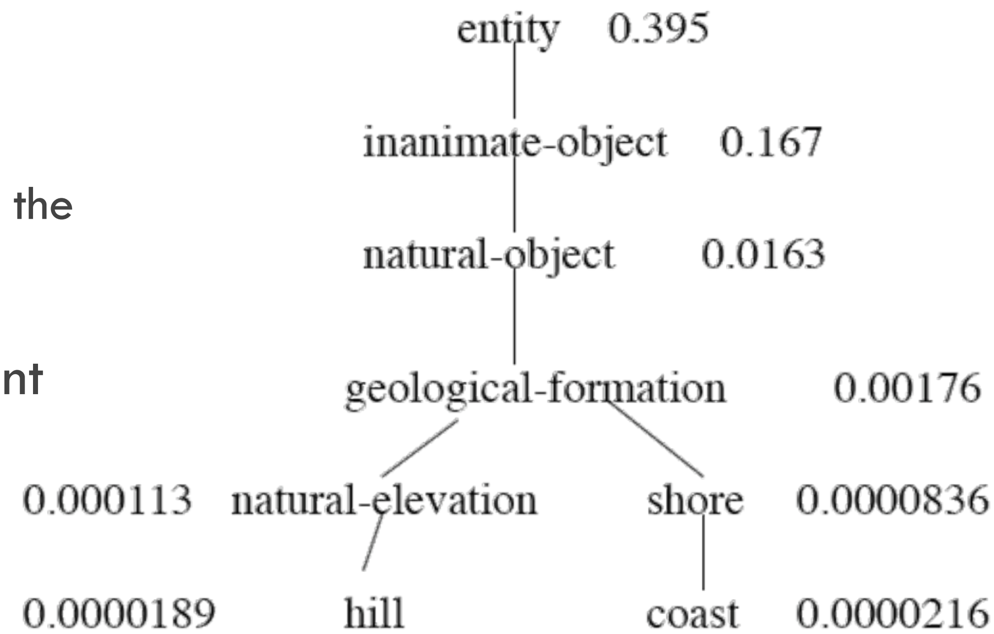
- $IC(c) = -\log P(c)$

Lowest common subsumer

- $LCS(c_1, c_2) =$ The most informative (lowest) node in the hierarchy subsuming both c_1 and c_2.

Resnik: The information content of the most informative subsumer of the two nodes:

- $sim_{resnik}(c_1, c_2) = -\log P(\,LCS(c_1, c_2)\,)$



Adapted from Dan Jurafsky (Stanford)

What about glosses? (Lesk '86)

Two concepts are similar if their glosses contain similar words

- *Drawing paper*: paper that is specially prepared for use in drafting
- *Decal*: the art of transferring designs from specially prepared paper to a wood or glass or metal surface

For each n–gram phrase that's in both glosses

- Add a score of n^2
- *Paper* and *specially prepared* for $1 + 2^2 = 5$
- Can extend to account for overlap in other relations (e.g., glosses of hypernyms and hyponyms)

Adapted from Dan Jurafsky (Stanford)

Thesaural Similarity

Thesaural relations admit a tree structure and typed relationships (and descriptions)

Use these properties to compute similarity.

Work to meld distributional, context-sensitive forms of similarity with thesaural versions to examine lexical semantics beyond local, lexical scope.

Adapted from Dan Jurafsky (Stanford)

Meaning Representation

Adopted from Chapter 17: The Representation of Meaning
NG Hwee Tou (CS4248 AY20/21 S1)

Desirable Properties of Meaning Representation Language

Our representation should have the following properties:

1. Verifiability
2. Unambiguous Representations
3. Canonical Form
4. Inference and Variables
5. Expressiveness

1. Verifiability

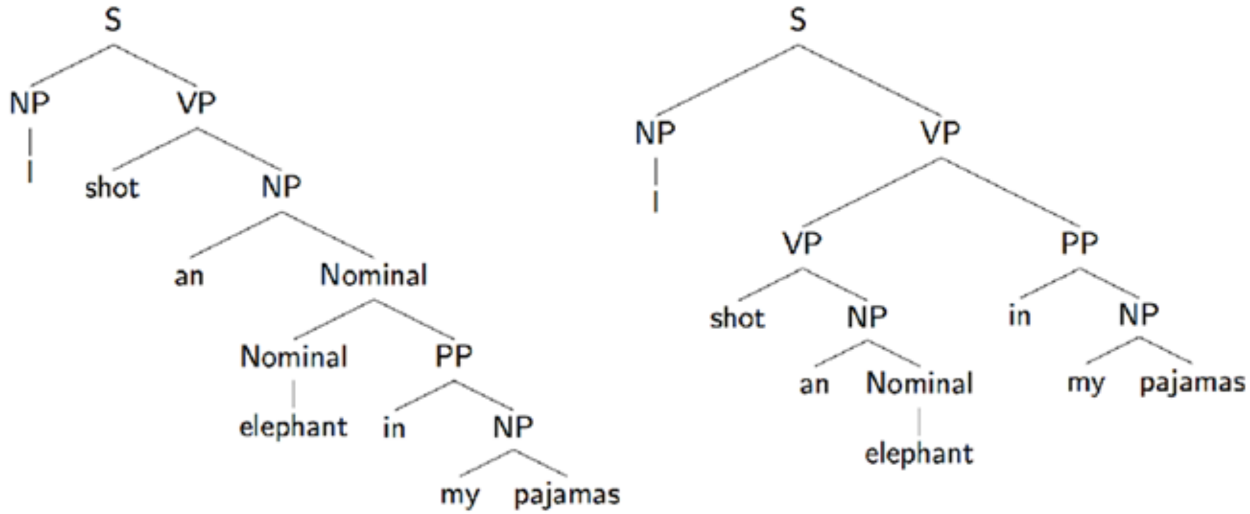
Able to determine the truth of the meaning representation of a sentence against the world modelled

That is, match the meaning representation of a sentence against representations of facts about the world modelled in the system's knowledge base

Does Maharani serve vegetarian food?
Serves(Maharani, VegetarianFood)

2. Unambiguous Representation

A meaning representation should be **unambiguous;**
each statement in a meaning representation should have one meaning



Syntax resolves some ambiguity

Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

3. Canonical Form

Different sentences with the same meaning should be assigned the same meaning representation:

Does Maharani serve vegetarian food?

3. Canonical Form

Different sentences with the same meaning should be assigned the same meaning representation:

Does Maharani serve vegetarian food?
Do they have vegetarian food at Maharani?
Are vegetarian dishes served at Maharani?
Does Maharani serve vegetarian fare?

- Different words (*dishes*, *food*, *fare*; *have*, *serve*)
- Different sentence structures (*Maharani has vegetarian food*, *They have vegetarian food at Maharani*)
- Having the same representation for these questions facilitates matching to facts in the knowledge base

3. Canonical Form

Assign the same meaning to different words

- *dishes, food, fare*

Assign the same meaning to different syntactic structures.
For example: Active and passive sentences

- *Maharani serves vegetarian dishes*
- *Vegetarian dishes are served by Maharani*

4. Inference and Variables

Inference: We should be able to draw valid inferences not explicitly stated in the meaning representation of the input sentence or program's knowledge base:

- *Maharani serves vegetarian food.*
- *Can vegetarians eat at Maharani?*
- *Need the inference rule that vegetarians eat vegetarian food*

Variables

- *Serves*(x, ***VegetarianFood***)

5. Expressiveness

Meaning representation language must be able to adequately express and represent the meaning of any natural language sentence.

First Order Logic

School of Computing

First-order logic (FOL)

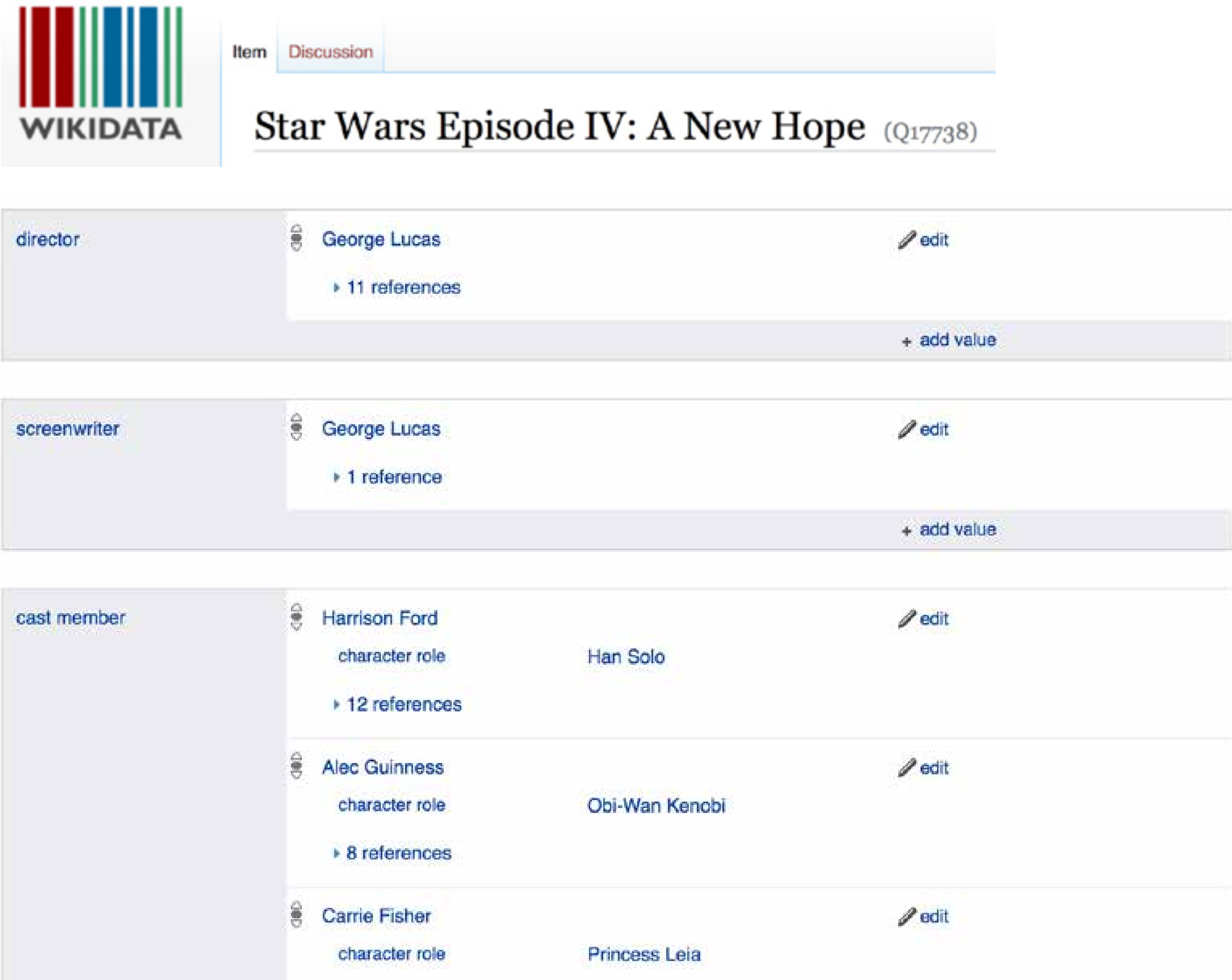
We want to represent every sentence as an unambiguous proposition in FOL.

Sentence	*Luke was fighting with Darth Vader*
FOL	FIGHT(LUKE, VADER)

FIGHT(LUKE, VADER)

This is a relation; we define what it means

These are constants; we know who they uniquely identify


WIKIDATA
Item
Discussion
Star Wars Episode IV: A New Hope (Q17738)
director
George Lucas
edit
11 references
add value
screenwriter
George Lucas
edit
1 reference
add value
cast member
Harrison Ford
edit
character role
Han Solo
12 references
Alec Guinness
edit
character role
Obi-Wan Kenobi
8 references
Carrie Fisher
edit
character role
Princess Leia

First-order logic (FOL)

How we map a natural language sentence to FOL is the task of semantic parsing; but we define the FOL relations and entities to be sensitive to what matters in our model.

Maybe in our Star Wars model, we want to preserve the difference between *fighting* and *dueling*.

Sentence	*Luke battled Vader*
Sentence	*Luke fought with Vader*
Sentence	*Luke was fighting with Darth Vader*
FOL	FIGHT(LUKE, VADER)

Sentence	*Skywalker dueled with Darth Vader*
FOL	DUEL(LUKE, VADER)

First-order logic (FOL)

How we map a natural language sentence to FOL is the task of semantic parsing; but we define the FOL relations and entities to be sensitive to what matters in our model.

For a robot model, there is only one possible "grabbing" kind of action, so subtle differences don't matter.

Sentence	*Grab the cup*
Sentence	*Snatch the cup!*
Sentence	*Take the cup*
FOL	GRAB(ROBOT, CUP)

First-order logic (FOL)

- Constants
- Relations (including properties)
- Variables
- Quantifiers
- Functions
- Logical connectives

Constants

Refer to specific entities in the world being described.

Refer to exactly one object in that world.

Dependent on the *model*; people, things, places, events, etc.

Constants

- DarthVader
- DarthMaul
- Emperor
- Luke
- HanSolo
- Leia
- Obi-Wan
- Chewbacca
- R2-D2
- C-3PO
- Yoda

- Luke_Lightsaber_1
- Luke_Lightsaber_2
- Vader_Lightsaber_1

- Tattoine
- Hoth
- Endor

The domain simply enumerates the objects; we know nothing else about them yet

Relations

N-ary relations hold among FOL terms (constants, variables, functions).

Informally, properties describe attributes of entities in the model (e.g. Luke being human) and are thus unary relations.

Unary (property)	HUMAN(LUKE), ROBOT(C-3PO)
binary relation	FIGHTS(LUKE, VADER)
ternary	GIVES(OBI-WAN, LUKE, LUKE_LIGHTSABER_1)
…	…

Properties

Formally, properties denote sets of elements in the domain

[[Human]] = {Darth Vader, Emperor, Luke, Han Solo, Leia, Obi-Wan}
[[Robot]] = {R2-D2, C-3PO}
[[Lightsaber]] = {Luke_Lightsaber_1, Luke_Lightsaber_2, Vader_Lightsaber_1}
[[Red]] = {Vader_Lightsaber_1}
[[Blue]] = {Luke_Lightsaber_1}
[[Green]] = {Luke_Lightsaber_2}
[[Planet]] = {Tattoine, Hoth, Endor}

Relations

Relations denote sets of tuples in the domain.

[[Fight]] = {<Darth Vader, Luke>, <Darth Vader, Obi-Wan>, <Luke, Emperor>}

[[Lives]] = {<Luke, Tattoine>}

[[Father]] = {<Darth Vader, Luke>, <Darth Vader, Leia>}

[[Gives]] = {<Obi-Wan, Luke, Luke_Lightsaber_1>}

Functions

OwnerOf(Luke_Lightsaber_1) yields *Luke*

What's the difference from a relation?

- A *function* takes *terms* as arguments, and results in another *term*, denoting an *object*.
- A *relation* takes terms as arguments, and results in a *sentence* that has a truth value.

Extensional definition

An extensional definition of a concept or term formulates its meaning by specifying its *extension*: *i.e.*, every object that falls under the definition of the concept in question.

- The denotation of RED is the set of objects that are lightsabers.
- The denotation of FIGHT is the set of pairs of entities that fight.

Logical connectives

Formal means for composing expressions in a meaning representation language (symbols, quantifiers)

Boolean operators on connectives yield known truth values

Negation	$\neg\varphi$	*True if φ is false*
Conjunction	$\varphi \wedge \psi$	*True if φ and ψ are both true*
Disjunction	$\varphi \vee \psi$	*True if φ or ψ is true*
Implication	$\varphi \Rightarrow \psi$	*True unless φ is true and ψ is false*
Equivalence	$\varphi \Leftrightarrow \psi$	*True if φ and ψ are both true or both false*

$\mathcal{M}$ star_wars

- Darth Vader
- Emperor
- Luke
- Han Solo
- Leia
- Obi-Wan
- Chewbacca
- R2-D2
- C-3PO

- Luke_Lightsaber_1
- Luke_Lightsaber_2
- Vader_Lightsaber

- Tattoine
- Hoth
- Endor

- [[HUMAN]] = {Darth Vader, Emperor, Luke, Han Solo, Leia, Obi-Wan}
- [[ROBOT]] = {R2-D2, C-3PO}

- [[LIGHTSABER]] = {Luke_Lightsaber_1, Luke_Lightsaber_2, Vader_Lightsaber_1}

- [[RED]] = {Vader_Lightsaber_1}
- [[BLUE]] = {Luke_Lightsaber_1}
- [[GREEN]] = {Luke_Lightsaber_2}

- [[PLANET]] = {Tattoine, Hoth, Endor}

- [[FIGHT]] = {<Darth Vader, Luke>, <Darth Vader, Obi-Wan>, <Luke, Emperor>}

- [[LIVES]] = {<Luke, Tattoine>}

- [[FATHER]] = {<Darth Vader, Luke>, <Darth Vader, Leia>}

- [[GIVES]] {<Obi-Wan, Luke, Luke_Lightsaber_1>}

Meaning

Model-theoretic semantics; the truth of a proposition is determined with respect to some model $\mathcal{M}$ of the world.

Separate from verification (does not need to be true of the real world); what would the world need to be like for the sentence to be true?

Denotation

We'll use $[[\cdot]]$ to describe the denotation of a term in a specific model $\mathcal{M}$.

$$[[\text{LUKE}]]_{\mathcal{M}}$$

$$[[\text{FIGHTS}]]_{\mathcal{M}}$$

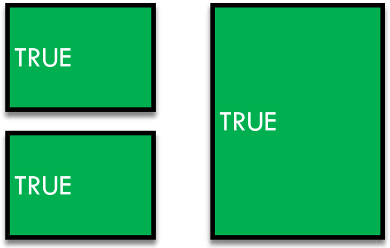
Truth

The truth of a proposition under a model depends on whether the denotation of the constants is in the denotation of the relation.

Luke lives on Tattoine	$([[\text{Luke}]]_{\mathcal{M}}, [[\text{Tattoine}]]_{\mathcal{M}}) \in [[\text{LIVES}]]_{\mathcal{M}}$	TRUE
Luke fights Han	$([[\text{Luke}]]_{\mathcal{M}}, [[\text{Han}]]_{\mathcal{M}}) \in [[\text{FIGHTS}]]_{\mathcal{M}}$	FALSE

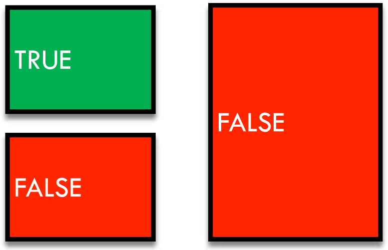
Luke lives on Tattoine and fights Darth Vader

$$([\![\text{Luke}]\!]\mathcal{M}, [\![\text{Tattoine}]\!]\mathcal{M}) \in [\![\textsc{Lives}]\!]\mathcal{M}$$
$$\wedge$$
$$([\![\text{Luke}]\!]\mathcal{M}, [\![\text{Darth Vader}]\!]\mathcal{M}) \in [\![\textsc{Fights}]\!]\mathcal{M}$$



Luke lives on Tattoine and fights Han

$$([\![\text{Luke}]\!]\mathcal{M}, [\![\text{Tattoine}]\!]\mathcal{M}) \in [\![\textsc{Lives}]\!]\mathcal{M}$$
$$\wedge$$
$$([\![\text{Luke}]\!]\mathcal{M}, [\![\text{Han}]\!]\mathcal{M}) \in [\![\textsc{Fights}]\!]\mathcal{M}$$



First-order logic

This machinery lets us evaluate the truth conditions about specific entities and relations in $\mathcal{M}$.

First-order logic

Does C-3PO fight anyone?

- Does C-3PO fight Luke?
- Does C-3PO fight Han?
- Does C-3PO fight Darth Vader?
- Does C-3PO fight Leia?
- Does C-3PO fight R2D2?
- Does C-3PO fight the Emperor?

Variables and Quantifiers

C-3PO fights someone

Variable

Fights(C-3PO, x)

x is a free variable with no committed assignment

Quantifier

$\exists x$ Fights(C-3PO, x)

x is now bound and has a truth value in $\mathcal{M}$

Existential and Universal Quantifiers

$\exists x$ Fights(C-3PO, x)

$\forall x$ Fights(C-3PO, x)

Semantic parsing

How do we get from a natural language statement to a proposition whose truth content we can evaluate in a model?

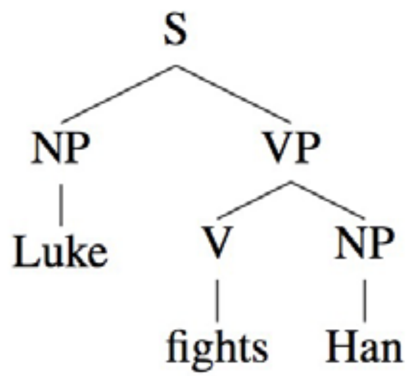
Luke fights Han

$$([[\text{Luke}]]_{\mathcal{M}}, [[\text{Han}]]_{\mathcal{M}}) \in [[\textsc{Fights}]]_{\mathcal{M}}$$

Compositionality

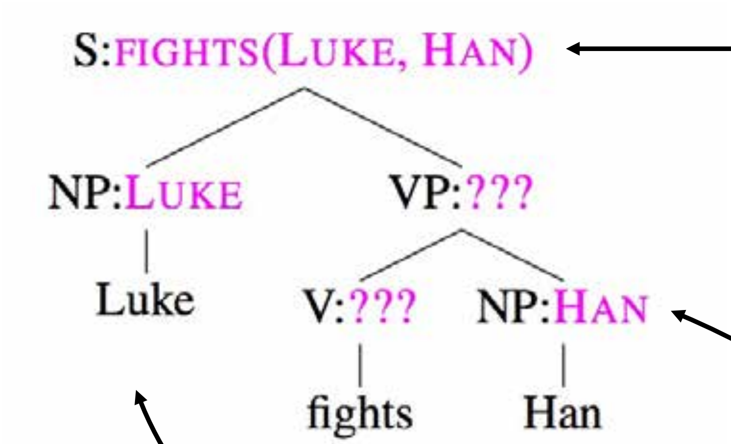
Principle of compositionality: the meaning of a complex expression is function of the meaning of its constituent parts (due to Frege).

"Constituent parts" = syntactic constituents



S.meaning = function(NP.meaning, VP.meaning)

How do we represent the meaning of partial structures?



This is the meaning we want in the end

The meaning of the entities is their denotation in $\mathcal{M}$

Lambda Calculus

Scheme all over again

School of Computing

Lambda calculus

Lambda expressions: descriptions of anonymous functions.

$$\lambda x \; P(x)$$

sequence of variables

FOL expression

Lambda calculus

Lambda expressions: descriptions of anonymous functions.

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

A lambda expression is a function that can be applied to FOL terms as arguments to yield new FOL expressions where the variables are bound to the argument terms.

Lambda calculus

Lambda expressions: descriptions of anonymous functions.

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

Argument

Yields:

$$P(A)$$

λ-reduction

Apply a lambda expression to a term. Unify them together.
Replace the λ variable by the term, then remove λx.

$\lambda y\ \lambda x$ NEAR(x, y)

The state of one thing being near something else

$\lambda y\ \lambda x$ NEAR(x, y)(CAFESTRADA)
λx NEAR$(x,$ CAFESTRADA)

The state of one thing being near Cafe Strada

λx NEAR$(x,$ CAFESTRADA)(Boalt)
NEAR(Boalt, CAFESTRADA)

Fully specified FOL formula: Boalt is near Cafe Strada

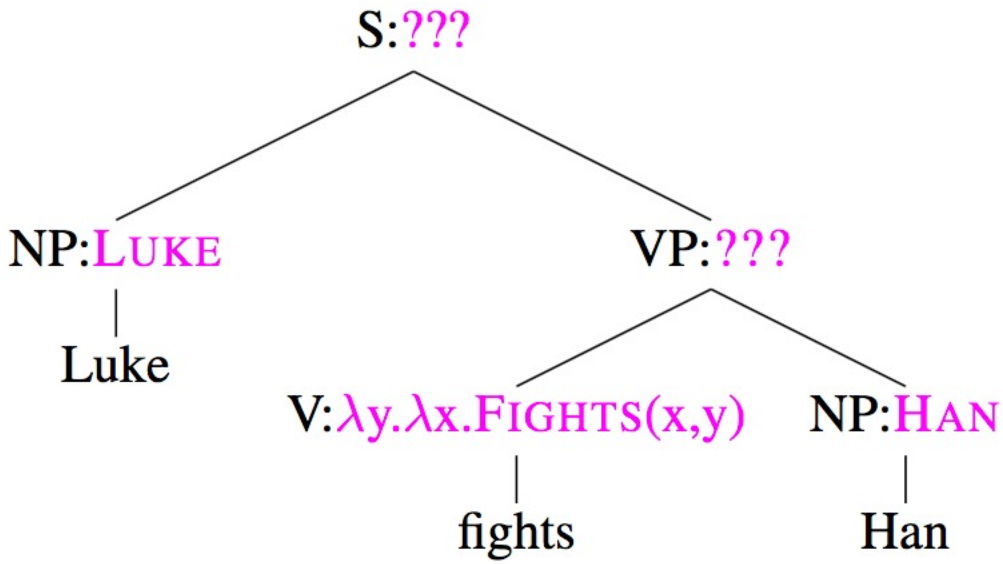
Lambda calculus

With lambda calculus, pieces of logical forms correspond to pieces of linguistic structure.

We can solve our problem by unifying each terminal with a entity (*Han*) or a lambda expression.

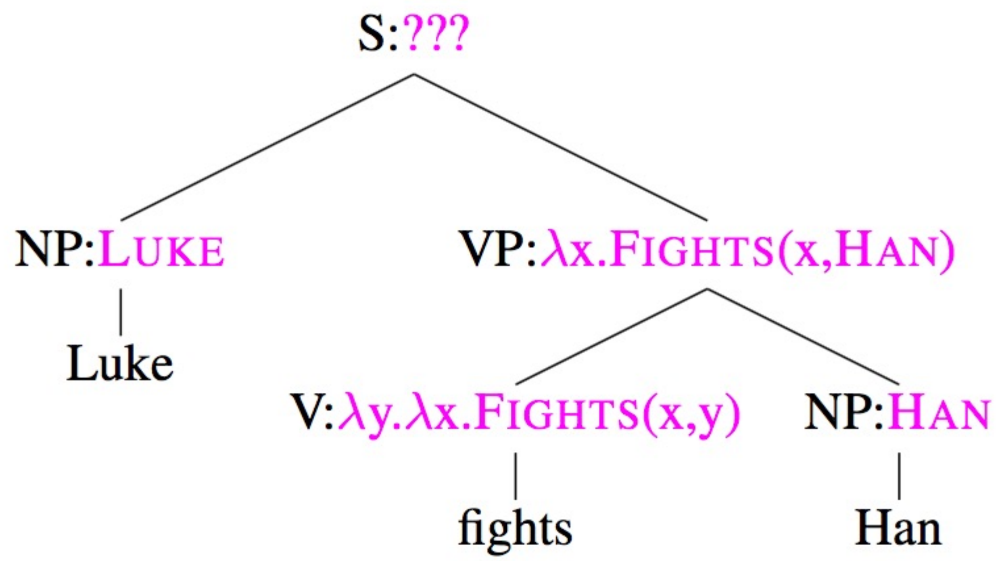
Fights is a verb that expects 2 roles: a subject argument (the fighter) and an object argument (the thing fought).

fights $\lambda y, \lambda x\ \textsc{Fights}(x, y)$



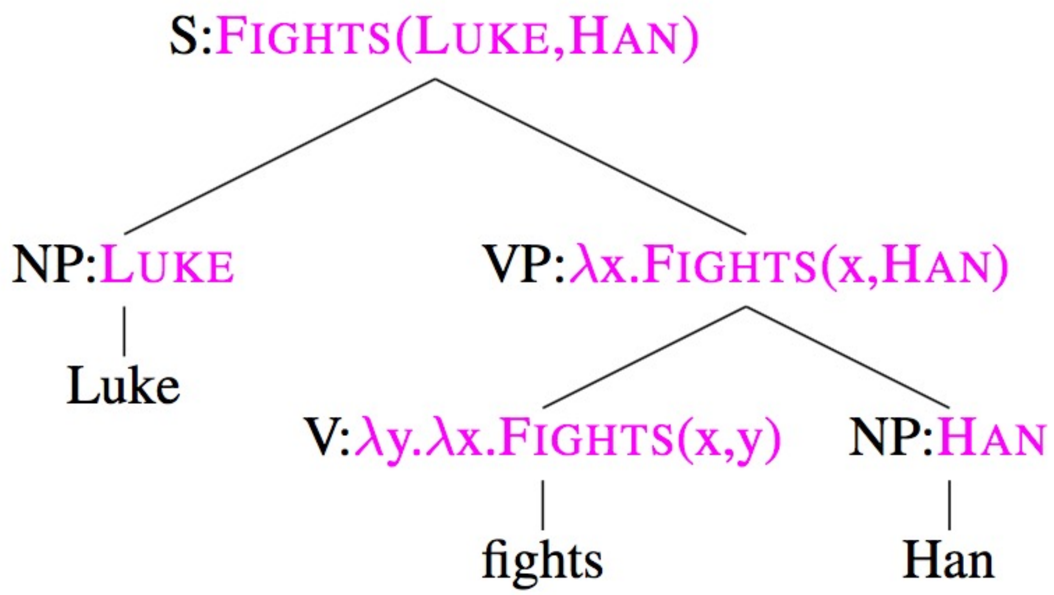
Here we can let naturally the NP to the right be the argument to this lambda expression to yield a new lambda expression

$$\lambda y, \lambda x\ \textsc{Fights}(x, y)(\textsc{Han}) \rightarrow \lambda x\ \textsc{Fights}(x, \textsc{Han})$$



Here we can state the NP to the left should be the argument to this lambda expression to yield a new lambda expression

$$\lambda x\ \text{FIGHTS}(x, \text{HAN})(\text{LUKE}) \rightarrow \text{FIGHTS}(\text{LUKE}, \text{HAN})$$



S:FIGHTS(LUKE,HAN)
NP:LUKE
VP:λx.FIGHTS(x,HAN)
Luke
V:λy.λx.FIGHTS(x,y)
NP:HAN
fights
Han

S $\longrightarrow$ NP VP	VP.sem NP.sem
VP $\longrightarrow$ V NP	V.sem NP.sem
V $\longrightarrow$ fights	$\lambda y, \lambda x$ Fights(x, y)
V $\longrightarrow$ battles	$\lambda y, \lambda x$ Fights(x, y)
NP $\longrightarrow$ Han	Han
NP $\longrightarrow$ Luke	Luke

In general, we can let the non-terminals determine how the semantics of their constituents are combined.

Adjuncts and Roles

How to treat categories?

Method 1: Create a unary predicate for each category

Lightsabers(Luke_Lightsaber_1)

OK! But we can't talk about *Lightsabers* (cf *Expressiveness* criteria)

❌ *MostDeadly(Luke_Lightsaber_1, Lightsabers)*

Predicate, not object!

Method 2 (Reification): Represent all concepts that we want to make statements about as full-fledged objects.

Is-A(Luke_Lightsaber_1, Lightsabers)
A-Kind-Of(Lightsabers, Weapon)

Make into a first class citizen

How would you express "*Luke gifts Rey a lightsaber*"?

Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

How would you express "*Luke gifts Rey a lightsaber*"?

$$\exists x \; Lightsaber(x) \wedge Gift(Luke, Rey, x)$$

How about: "*Yesterday, Luke gifted Rey a lightsaber reluctantly*"?

Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

Arguments and Adjuncts

How would you express "*Luke gifts Rey a lightsaber*"?

$$\exists x\ Lightsaber(x) \land Gift(Luke, Rey, x)$$

How about: "*Yesterday, Luke gifted Rey a lightsaber reluctantly*"?

$$\exists x\ Lightsaber(x) \land Gift(Luke, Rey, x, Yesterday, Reluctantly)$$

One option: extend the arity of the relation (require more arguments).

But that's not great because we need a separate predicate for every possible combination of arguments (even for those that aren't required).

Adjuncts (optional) in linguistics, as opposed to obligatory arguments.

Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

We can reify the event to an existentially quantified variable of its own, and then use it as an argument in other relations.

$$
\begin{aligned}
&\exists e, x\ \mathit{GiveEvent}(e) \\
&\land\ \mathit{Giver}(e,\ \mathit{Luke}) \\
&\land\ \mathit{Gift}(e, x) \\
&\land\ \mathit{Lightsaber}(x) \\
&\land\ \mathit{Recipient}(e,\ \mathit{Rey}) \\
&\land\ \mathit{Time}(e,\ \mathit{Yesterday}) \\
&\land\ \mathit{Manner}(e,\ \mathit{Reluctanctly})
\end{aligned}
$$

Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

Generalized Roles

NUS National University of Singapore | School of Computing

$\exists e, y$ BREAK(e)
$\wedge$ AGENT(e,Leia)
$\wedge$ THEME(e, y)
$\wedge$ "window"(y)

$\exists e, y$ OPEN(e)
$\wedge$ AGENT(e,Luke)
$\wedge$ THEME(e, y)
$\wedge$ "door"(y)

$\exists e, y$ BREAKING-EVENT(e)
$\wedge$ BREAKER(e,Leia)
$\wedge$ BROKEN-THING(e, y)
$\wedge$ WINDOW(y)

$\exists e, y$ OPENING-EVENT(e)
$\wedge$ OPENER(e,Luke)
$\wedge$ OPENED-THING(e, y)
$\wedge$ DOOR(y)

These predicates have a lot in common: direct causal responsibility for the events, have volition, often animate

Adopted from Lecture 18: Semantic Roles (March 31, 2020)
David Bamman (UC Berkeley Info 159/259)

Summary

Lexical Semantics

- Meaning of and relationships among words
- WordNet 3.0 hand-annotated many of these relationships
- Compute relatedness via ontological tree structure

Logical Semantics

- Meaning representation with FOL with terms and relations
- Unify (fill expected arguments appropriately) while parsing