Natural Language Processing (CSE 517): Predicate-Argument Semantics

Noah Smith
© 2018

University of Washington
nasmith@cs.washington.edu

May 16, 2018
Semantics vs. Syntax

Syntactic theories and representations focus on the question of which strings in $\mathcal{V}^+$ are in the language.

Semantics is about understanding what a string in $\mathcal{V}^+$ means.

Sidestepping a lengthy and philosophical discussion of what “meaning” is, we’ll consider two meaning representations:

- Predicate-argument structures, also known as event frames (today)
- Truth conditions represented in first-order logic (next time)
Motivating Example: Who did What to Who(m)?

- Warren bought the stock.
- They sold the stock to Warren.
- The stock was bought by Warren.
- The purchase of the stock by Warren surprised no one.
- Warren’s stock purchase surprised no one.
Motivating Example: Who did What to Who(m)?

- Warren bought the stock.
- They sold the stock to Warren.
- The stock was bought by Warren.
- The purchase of the stock by Warren surprised no one.
- Warren’s stock purchase surprised no one.
Motivating Example: Who did What to Who(m)?

- Warren bought the stock.
- They sold the stock to Warren.
- The stock was bought by Warren.
- The purchase of the stock by Warren surprised no one.
- Warren’s stock purchase surprised no one.
Motivating Example: Who did What to Who(m)?

- Warren bought the stock.
- They sold the stock to Warren.
- The stock was bought by Warren.
- The purchase of the stock by Warren surprised no one.
- Warren’s stock purchase surprised no one.
Motivating Example: Who did What to Who(m)?

- Warren bought the stock.
- They sold the stock to Warren.
- The stock was bought by Warren.
- The purchase of the stock by Warren surprised no one.
- Warren’s stock purchase surprised no one.

In this buying/purchasing event/situation, Warren played the role of the buyer, and there was some stock that played the role of the thing purchased.
Motivating Example: Who did What to Who(m)?

- Warren bought the stock.
- They sold the stock to Warren.
- The stock was bought by Warren.
- The purchase of the stock by Warren surprised no one.
- Warren’s stock purchase surprised no one.

In this buying/purchasing event/situation, Warren played the role of the buyer, and there was some stock that played the role of the thing purchased.

Also, there was presumably a seller, only mentioned in one example.
Motivating Example: Who did What to Who(m)?

- Warren bought the stock.
- They sold the stock to Warren.
- The stock was bought by Warren.
- The purchase of the stock by Warren surprised no one.
- Warren’s stock purchase surprised no one.

In this buying/purchasing event/situation, Warren played the role of the buyer, and there was some stock that played the role of the thing purchased.

Also, there was presumably a seller, only mentioned in one example.

In some examples, a separate “event” involving surprise did not occur.
Semantic Roles: Breaking

- Jesse broke the window.
- The window broke.
- Jesse is always breaking things.
- The broken window testified to Jesse’s malfeasance.
Semantic Roles: Breaking

- Jesse broke the window.
- The window broke. ?
- Jesse is always breaking things.
- The broken window testified to Jesse’s malfeasance.

A breaking event has a Breaker and a Breakee.
Semantic Roles: Eating

- Eat!
- We ate dinner.
- We already ate.
- The pies were eaten up quickly.
- Our gluttony was complete.
Semantic Roles: Eating

- **Eat! (you, listener)**
- **We ate dinner.**
- **We already ate.**
- **The pies were eaten up quickly.**
- **Our gluttony was complete.**

An eating event has an **EATER** and **FOOD**, neither of which needs to be mentioned explicitly.
Abstraction?

$\text{Breaker} \neq \text{Eater}$
Abstraction?

\textbf{Breaker} \neq \textbf{Eater}

Both are actors that have some causal responsibility for changes in the world around them.
Abstraction?

\[ \text{Breaker} \equiv \text{Eater} \]

Both are actors that have some causal responsibility for changes in the world around them.

\[ \text{Breakee} \equiv \text{Food} \]
Abstraction?

\[
\text{Breaker} \equiv \text{Eater}
\]

Both are actors that have some causal responsibility for changes in the world around them.

\[
\text{Breakee} \equiv \text{Food}
\]

Both are greatly affected by the event, which “happened to” them.
**Thematic Roles**

(Jurafsky and Martin, 2015, with modifications)

**AGENT**  
The waiter spilled the soup.

**EXPERIENCER**  
John has a headache.

**FORCE**  
The wind blows debris from the mall into our yards.

**THEME**  
Jesse broke the window

**RESULT**  
The city built a regulation-size baseball diamond.

**CONTENT**  
Mona asked, “You met Mary Ann at a supermarket?”

**INSTRUMENT**  
He poached catfish, stunning them with a shocking device.

**BENEFICIARY**  
Ann Callahan makes hotel reservations for her boss.

**SOURCE**  
I flew in from Boston.

**GOAL**  
I drove to Portland.
Verb Alternation Examples: Breaking and Giving

Breaking:
- Agent/subject; Theme/object; Instrument/PP
- Instrument/subject; Theme/object
- Theme/subject

Giving:
- Agent/subject; Beneficiary/object; Theme/second-object
- Agent/subject; Theme/object; Beneficiary/PP

Levin (1993) codified English verbs into classes that share patterns (e.g., verbs of throwing: throw/kick/pass).
Fillmore (1968), among others, argued for semantic roles in linguistics.
Remarks

- Fillmore (1968), among others, argued for semantic roles in linguistics.
- By now, it should be clear that the expressiveness of NL (at least English) makes semantic analysis rather distinct from syntax.
Fillmore (1968), among others, argued for semantic roles in linguistics.

By now, it should be clear that the expressiveness of NL (at least English) makes semantic analysis rather distinct from syntax.

General challenges to analyzing semantic roles:

- What are the predicates/events/frames/situations?
- What are the roles/participants for each one?
- What algorithms can accurately identify and label all of them?
Semantic Role Labeling

Input: a sentence $x$

Output:

- A collection of predicates, each consisting of:
  - a label, sometimes called the frame
  - a span
  - a set of arguments, each consisting of:
    - a label, usually called the role
    - a span

In principle, spans might have gaps, though in most conventions they usually do not.
The Importance of Lexicons

Like syntax, any annotated dataset is the product of extensive development of conventions.

Many conventions are specific to particular words, and this information is codified in structured objects called **lexicons**.

You should think of every semantically annotated dataset as both the data and the lexicon.

We consider two examples.
Frames are verb senses (later extended, though)

Lexicon maps verb-sense-specific roles onto a small set of abstract roles (e.g., Arg0, Arg1, etc.)

Annotated on top of the Penn Treebank, so that arguments are always constituents.
fall.01 (move downward)

- **ARG1**: logical subject, patient, thing falling
- **ARG2**: extent, amount fallen
- **ARG3**: starting point
- **ARG4**: ending point
- **ARGM-LOC**: medium

- Sales fell to $251.2 million from $278.8 million.
- The average junk bond fell by 4.2%.
- The meteor fell through the atmosphere, crashing into Palo Alto.
fall.01 (move downward)

- **ARG1**: logical subject, patient, thing falling
- **ARG2**: extent, amount fallen
- **ARG3**: starting point
- **ARG4**: ending point
- **ARGM-LOC**: medium

- **Sales** fell to $251.2 million from $278.8 million.
- **The average junk bond** fell by 4.2%.
- **The meteor** fell through the atmosphere, crashing into Palo Alto.
fall.01 (move downward)

- **ARG1**: logical subject, patient, thing falling
- **ARG2**: extent, amount fallen
- **ARG3**: starting point
- **ARG4**: ending point
- **ARGM-LOC**: medium

- **Sales** fell to $251.2 million from $278.8 million.
- The average junk bond fell by 4.2%.
- The meteor fell through the atmosphere, crashing into Palo Alto.
fall.01 (move downward)

- **ARG1**: logical subject, patient, thing falling
- **ARG2**: extent, amount fallen
- **ARG3**: starting point
- **ARG4**: ending point
- **ARGIN-LOC**: medium

- *Sales* fell to $251.2 million from $278.8 million.
- The average junk bond fell by 4.2%.
- The meteor fell through the atmosphere, crashing into Palo Alto.
fall.01 (move downward)

- **ARG1**: logical subject, patient, thing falling
- **ARG2**: extent, amount fallen
- **ARG3**: starting point
- **ARG4**: ending point
- **ARGM-LOC**: medium

- **Sales** fell to $251.2 million from $278.8 million.
- The average junk bond fell by 4.2%.
- The meteor fell through the atmosphere, crashing into Palo Alto.
fall.01 (move downward)

- **ARG1**: logical subject, patient, thing falling
- **ARG2**: extent, amount fallen
- **ARG3**: starting point
- **ARG4**: ending point
- **ARGM-LOC**: medium

- Sales fell to $251.2 million from $278.8 million.
- The average junk bond fell by 4.2%.
- The meteor fell through the atmosphere, crashing into Palo Alto.
World Bank president Paul Wolfowitz has fallen back on his last resort.
fall.08 (fall back, rely on in emergency)

- **ARG0**: thing falling back
- **ARG1**: thing fallen back on

- World Bank president Paul Wolfowitz has fallen back on his last resort.
World Bank president Paul Wolfowitz has fallen back on his last resort.
Many people keep falling for the idea that lowering taxes on the rich benefits everyone.
Many people keep falling for the idea that lowering taxes on the rich benefits everyone.
fall.10 (fall for a trick; be fooled by)

ARG1: the fool
ARG2: the trick

Many people keep falling for the idea that lowering taxes on the rich benefits everyone.
Frames can be any content word (verb, noun, adjective, adverb)
About 1,000 frames, each with its own roles
Both frames and roles are hierarchically organized
Annotated without syntax, so that arguments can be anything

https://framenet.icsi.berkeley.edu
change_position_on_a_scale

- **ITEM**: entity that has a position on the scale
- **ATTRIBUTE**: scalar property that the **ITEM** possesses
- **DIFFERENCE**: distance by which an **ITEM** changes its position
- **FINAL_STATE**: **ITEM**’s state after the change
- **FINAL_VALUE**: position on the scale where **ITEM** ends up
- **INITIAL_STATE**: **ITEM**’s state before the change
- **INITIAL_VALUE**: position on the scale from which the **ITEM** moves
- **VALUE_RANGE**: portion of the scale along which values of **ATTRIBUTE** fluctuate
- **DURATION**: length of time over which the change occurs
- **SPEED**: rate of change of the value
- **GROUP**: the group in which an **ITEM** changes the value of an **ATTRIBUTE**
Attacks on civilians **decreased** over the last four months
change_position_on_a_scale

The **Attribute** is left unfilled but is understood from context (i.e., “frequency”).
Verbs: advance, climb, decline, decrease, diminish, dip, double, drop, dwindle, edge, explode, fall, fluctuate, gain, grow, increase, jump, move, mushroom, plummet, reach, rise, rocket, shift, skyrocket, slide, soar, swell, swing, triple, tumble

Nouns: decline, decrease, escalation, explosion, fall, fluctuation, gain, growth, hike, increase, rise, shift, tumble

Adverb: increasingly
change_position_on_a_scale

(birth_scenario also inherits from sexual_reproduction_scenario.)
The paper that started it all: Gildea and Jurafsky (2002) used FrameNet lexicon (which includes prototypes, not really a corpus).

- When FrameNet started releasing corpora, the task was reformulated. Example open-source system: SEMAFOR (Das et al., 2014).
Semantic Role Labeling Tasks

The paper that started it all: Gildea and Jurafsky (2002) used FrameNet lexicon (which includes prototypes, not really a corpus).

► When FrameNet started releasing corpora, the task was reformulated. Example open-source system: SEMAFOR (Das et al., 2014).

The PropBank corpus is used directly for training/testing.
Semantic Role Labeling Tasks

The paper that started it all: Gildea and Jurafsky (2002) used FrameNet lexicon (which includes prototypes, not really a corpus).

▶ When FrameNet started releasing corpora, the task was reformulated. Example open-source system: SEMAFOR (Das et al., 2014).

The PropBank corpus is used directly for training/testing.


▶ In 2008 and 2009, the task was cast as a kind of dependency parsing.
▶ In 2009, seven languages were included in the task.
Methods

Boils down to labeling spans (with frames and roles).

It’s mostly about features.
The San Francisco Examiner issued a special edition around noon yesterday.
Example: Path Features

Path from The San Francisco Examiner to issued: NP↑S↓VP↓VBD
Example: Path Features

```
NP-SBJ
  /   \\
/     \                  
DT    NNP    NNP    NNP
The   San    Francisco    Examiner
       
VP
  /   \\
/     \                  
VBD       NP
issued

PP-TMP
  /   \\
/     \                  
DT    JJ    NN    IN    NN    NN    NP-TMP
a    special    edition    around    noon    NN

Path from a special edition to issued: NP↑VP↓VBD
```
The span-labeling decisions interact a lot!

- Presence of a frame increases the expectation of certain roles
- Roles for the same predicate shouldn’t overlap
- Some roles are mutually exclusive or require each other (e.g., “resemble”)
Methods: Beyond Features

The span-labeling decisions interact a lot!

- Presence of a frame increases the expectation of certain roles
- Roles for the same predicate shouldn’t overlap
- Some roles are mutually exclusive or require each other (e.g., “resemble”)

Ensuring well-formed outputs:

- Using syntax as a scaffold allows efficient prediction; you’re essentially labeling the parse tree (Toutanova et al., 2008).
- Others have formulated the problem as constrained, discrete optimization (Punyakanok et al., 2008).
- Also greedy methods (Björkelund et al., 2010) and joint methods for syntactic and semantic dependencies (Henderson et al., 2013).
Methods: Beyond Features

The span-labeling decisions interact a lot!

- Presence of a frame increases the expectation of certain roles
- Roles for the same predicate shouldn’t overlap
- Some roles are mutually exclusive or require each other (e.g., “resemble”)

Ensuring well-formed outputs:

- Using syntax as a scaffold allows efficient prediction; you’re essentially labeling the parse tree (Toutanova et al., 2008).
- Others have formulated the problem as constrained, discrete optimization (Punyakanok et al., 2008).
- Also greedy methods (Björkelund et al., 2010) and joint methods for syntactic and semantic dependencies (Henderson et al., 2013).

Current work:

- Some recent attempts to merge FrameNet and PropBank have shown promise (FitzGerald et al., 2015; Kshirsagar et al., 2015)
Related Problems in “Relational” Semantics

- **Coreference resolution**: which mentions (within or across texts) refer to the same entity or event?
- **Entity linking**: ground such mentions in a structured knowledge base (e.g., Wikipedia)
- **Relation extraction**: characterize the relation among specific mentions

**Information extraction**: transform text into a structured knowledge representation

- Classical IE starts with a predefined schema
- “Open” IE includes the automatic construction of the schema; see http://ai.cs.washington.edu/projects/open-information-extraction
General Remarks

Criticisms of semantic role labeling:

- Semantic roles are just “syntax++” since they don’t allow much in the way of reasoning (e.g., question answering).
Criticisms of semantic role labeling:

- Semantic roles are just "syntax++" since they don’t allow much in the way of reasoning (e.g., question answering).
- Lexicon building is slow and requires expensive expertise. Can we do this for every (sub)language?
General Remarks

Criticisms of semantic role labeling:

▶ Semantic roles are just “syntax++” since they don’t allow much in the way of reasoning (e.g., question answering).
▶ Lexicon building is slow and requires expensive expertise. Can we do this for every (sub)language?

We’ve now had a taste of two branches of semantics:

▶ Lexical semantics (e.g., supersense tagging)
▶ Relational semantics (e.g., semantic role labeling)
General Remarks

Criticisms of semantic role labeling:

- Semantic roles are just “syntax++” since they don’t allow much in the way of reasoning (e.g., question answering).
- Lexicon building is slow and requires expensive expertise. Can we do this for every (sub)language?

We’ve now had a taste of two branches of semantics:

- Lexical semantics (e.g., supersense tagging)
- Relational semantics (e.g., semantic role labeling)

Next up, a third:

- Compositional semantics
dragonfly • conveyor belt • finger food • anteater • brain teaser • C++ code • leather belt • birthday • Batman • firehose • fish food • steel wool • jazz musician • staple remover • fisheye • Cookie Monster • Spanish teacher • computer science • student teacher • U.S. Constitution • Facebook status • coffee cake • iron fist • Toy Story • glue gun • baby food • Labor Day • thesis supervisor • flyswatter • dawn raid • paper clip • surge protector • project team • spaghetti monster • tomato sauce • string orchestra • rubber duck • piano key • toothbrush • heartburn • Shannon entropy • elevator button

Your job is to group these into categories and explain those categories to the class; focus on the semantic relationship between the two nouns in each compound. You may wish to think of other compounds to help make your case.
References


