LING 520: Computational Analysis of English

Semester: FALL '16

Instructor: Sowmya Vajjala

Iowa State University, USA

8 November 2016

Class Outline

- Regular expressions on parse trees: references
- Semantics: Relevance in NLP
- Overview of tools and resources in computational semantics
- NLTK Exercises
- Reminder 1: Assignment 5 is due on 15th November.
- Reminder 2: Those who did not submit their 1 page reports - Please talk to me after the class.

Regular Expressions on Parse Trees

- It is possible to create regular expressions with parse trees (example purpose: patterns of parent-child relationship between nodes)
- Tregex is a utility that can work with Stanford parser trees, available for download on Stanford NLP website. http://nlp.stanford.edu/software/tregex.shtml
- has a UI that should work on all computers.
- Tregex also comes with another utility called Tsurgeon, which allows us to edit parse trees in locations matched by a tregex pattern.

Regular Expressions on Parse Trees

- It is possible to create regular expressions with parse trees (example purpose: patterns of parent-child relationship between nodes)
- Tregex is a utility that can work with Stanford parser trees, available for download on Stanford NLP website. http://nlp.stanford.edu/software/tregex.shtml
- has a UI that should work on all computers.
- Tregex also comes with another utility called Tsurgeon, which allows us to edit parse trees in locations matched by a tregex pattern.
- Very useful if you want to extract some syntactic patterns from parse trees (e.g., "clause" is not a part of constituency trees. But you can extract clause patterns from parse trees using tregex).

Semantics: Relevance in NLP

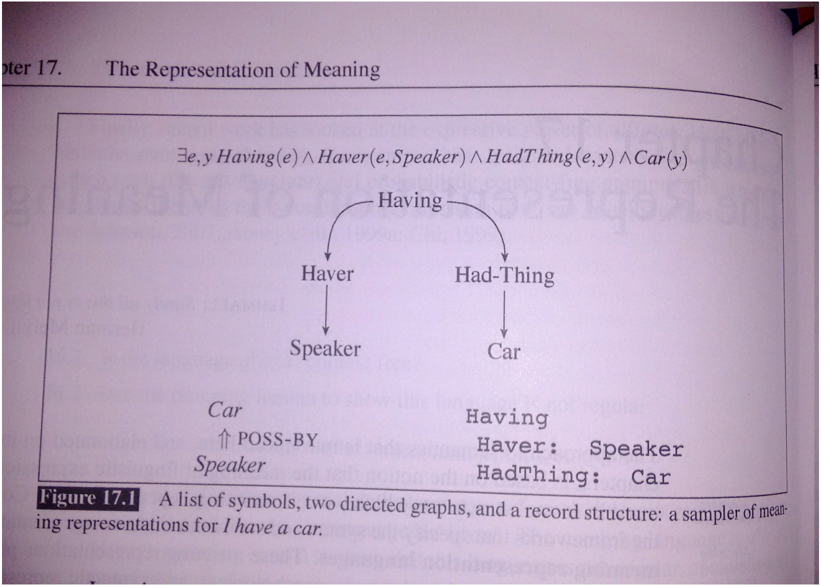
Semantics is useful in artificial intelligence tasks such as:

- Making a robot follow written set of instructions
- Finding if a student answer is relevant to the question asked
- Holding a dialogue with user (real dialogue, not like Eliza)
- Making the computer understand humor, sarcasm, metaphor etc.
- Tasks like sentiment analysis too may eventually require semantics to understand what the review really means.

How is semantics represented in NLP?

- Semantics is studied in using formal systems of logic such as first order logic.
- Computational semantics follows that same tradition.
- I am not going into that theoretical details (again, they are taught in separate courses!)
- If you are primarily a linguist (not an applied linguist) and are trained in logic, you can go through Chapter 10 in NLTK for practical stuff and Chapters 17–21 for a more rigorous treatment of computational semantics.
- For the state of the art on research in this topic, view this tutorial: http://yoavartzi.com//pub/afz-tutorial.acl.2013.pdf

How is semantics represented in NLP?



pter 17. The Representation of Meaning

$\exists e,y\ Having(e) \wedge Haver(e,Speaker) \wedge HadThing(e,y) \wedge Car(y)$

Having

Haver

Had-Thing

Speaker

Car

Car

⇑ POSS-BY

Speaker

```
Having
 Haver:   Speaker
 HadThing:  Car
```

Figure 17.1 A list of symbols, two directed graphs, and a record structure: a sampler of meaning representations for *I have a car.*

Semantics in NLP

- Lexical semantics - understanding the meaning of words
- Distributional semantics: ”a word is characterized by the company it keeps” (J.R.Firth)
- Syntax driven semantics - understanding the meaning of sentences

Semantics of words: Lexical Semantics in NLP

- identifying word sense in context (ambiguous words like bank, for example)
- identifying different semantic relations between words (eg: synonymy, antonymy, hyponymy, hypernymy, homonymy, polysemy, metonymy, meronymy)
- identifying different semantic relations between words based on how they are "used" in real word (distributional semantics)

Relations between words: Examples

- synonymy: good and nice can be considered synonyms
- antonymy: good and bad are antonyms
- hyponymy and hypernymy: crow is a hyponym of bird. Bird is a hypernym of crow.
- homonymy: words having identical form but different meaning
- polysemy: one word having multiple meanings (differences between homonymy and polysemy.. some examples: https://goo.gl/1bU1D0)
- metonymy: usage of an expression to indicate something closely associated with it. Using Whitehouse to refer to the government administration.
- meronym: part-of-whole relationships. NLP is a meronym of AI. keys are meronyms of keyboard etc.

Resources for Computational Lexical Semantics

- Wordnet : database of lexical semantic relations between English words. (exists for some other languages as well) https://wordnet.princeton.edu/
- VerbNet: a resource for English verbs, based on Levin's verb classes. Links to wordnet. (Beth Levin, the linguist) http://verbs.colorado.edu/~mpalmer/projects/verbnet.html
- FrameNet: based on linguistic theories on "frame semantics", which uses something called "frame" to describe a event. For example, the frame about "food" may have elements: cook, thing being cooked, source of heat etc. (https://framenet.icsi.berkeley.edu/fndrupal/about)
- Typically, such resources are used to do word sense disambiguation, and for deriving information about similarity between words (useful in various NLP tasks including essay grading)

.. resources are starting to emerge for other languages as well. But most work is in English.

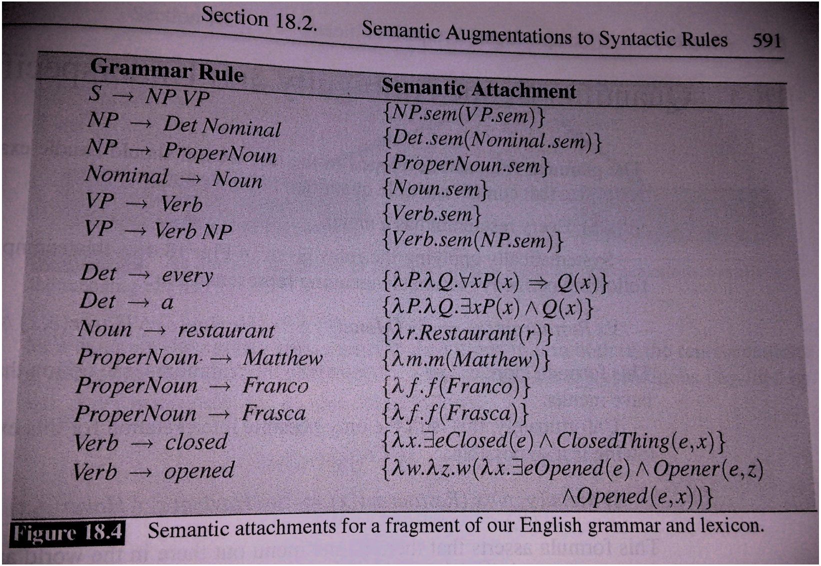
Semantics of words: Distributional Semantics in NLP

- Idea: derive information about semantic similarity of words by studying their usage in context in large corpora
- Example: if I have several sentences about pets, after applying some distributional semantics method, a computer should understand dogs, cats, horses etc are all pets people have.
- Use: particularly useful for covering words not covered by resources like wordnet. Also useful for doing NLP with languages that do not have such fancy resources.
- How?: Collecting contexts of appearance of words (n words before and/or after a given word), calculating what other words have similar contexts.
- typical methods: latent semantic analysis, word embeddings etc (discussed in detail in 515)
- Python has libraries that can do distributional semantics.

Syntax driven semantic analysis

- idea: meaning of a sentence is not purely based on meaning of the words in it, but on the ordering and grouping of words and relations between them.
- that is, meaning needs syntactic structure.
- Process: augment a CFG with semantic rules that specify how to get the meaning representation based on syntax.

Semantics enhanced parsing rules



Grammar Rule	Semantic Attachment
$S \rightarrow NP\ VP$	$\{NP.sem(VP.sem)\}$
$NP \rightarrow Det\ Nominal$	$\{Det.sem(Nominal.sem)\}$
$NP \rightarrow ProperNoun$	$\{ProperNoun.sem\}$
$Nominal \rightarrow Noun$	$\{Noun.sem\}$
$VP \rightarrow Verb$	$\{Verb.sem\}$
$VP \rightarrow Verb\ NP$	$\{Verb.sem(NP.sem)\}$
$Det \rightarrow every$	$\{\lambda P.\lambda Q.\forall x P(x) \Rightarrow Q(x)\}$
$Det \rightarrow a$	$\{\lambda P.\lambda Q.\exists x P(x) \land Q(x)\}$
$Noun \rightarrow restaurant$	$\{\lambda r.Restaurant(r)\}$
$ProperNoun \rightarrow Matthew$	$\{\lambda m.m(Matthew)\}$
$ProperNoun \rightarrow Franco$	$\{\lambda f.f(Franco)\}$
$ProperNoun \rightarrow Frasca$	$\{\lambda f.f(Frasca)\}$
$Verb \rightarrow closed$	$\{\lambda x.\exists e Closed(e) \land ClosedThing(e,x)\}$
$Verb \rightarrow opened$	$\{\lambda w.\lambda z.w(\lambda x.\exists e Opened(e) \land Opener(e,z)$ $\land Opened(e,x))\}$

Figure 18.4 Semantic attachments for a fragment of our English grammar and lexicon.

Meaning of sentences

Semantically parsing sentences.

- https://goo.gl/9nltNU
- http://cogcomp.cs.illinois.edu/page/demo_view/srl
- http://barbar.cs.lth.se:8081/parse
- Resource: PropBank - resource of sentences annotated with semantic roles (English and Chinese) https://www.researchgate.net/figure/283893596_fig3_Fig-19-Sample-PropBank-entry

Exercises

Exercises I am going to give today and from now on will be more open-ended and undirected, for two reasons:

1. We are into Week 12, and I want you to "think" more rather than doing what I tell you to do.
2. I don't know what specific problems you are interested in and where do these issues like semantics come into picture in that.

Exercise 1: NLTK exercises

- Chapter 3, Section 5 in NLTK demonstrates what we can do with WordNet in NLTK.
- Practice the examples there, and come up with problems in your topic of interest where you may find such a resource useful.
- In about 20 minutes, I want some of you to comment on what scenarios you thought about for using wordnet.

Exercise 2: using Tregex

- Download tregex, check if you can use the interface and run the program without problems.
- Following the power point slides on Stanford Tregex page, learn to use the tool to extract patterns from trees.
- If you are interested, read this 2006 short article that describes these two tools for the first time: http://nlp.stanford.edu/pubs/levy_andrew_lrec2006.pdf
- Figure out if you can use it in Python code somehow.