GENERATIVE LEXICON

Kevin Duh
NAIST Grammar Study Group
6/27/2013

References

- The material here closely follows:
  - J. Pustejovsky (1991). The Generative Lexicon, *in Computational Linguistics, vol. 17(4).*
  - http://acl.ldc.upenn.edu/J/J91/J91-4003.pdf

- Other useful references:
  - Recent introduction: http://www.cs.brandeis.edu/~jamesp/classes/LING130/ELS-GL-Entry.pdf
  - Book: http://mitpress.mit.edu/books/generative-lexicon

Main Ideas of the Generative Lexicon (GL)

- The lexicon is not a fixed thing, but is creative and generative (as is syntax)
  - How we use finite number of words in infinite number of contexts
- GL focuses on the distributed nature of semantic compositionality, e.g.
  - Meanings of “fast”:
    - Fast(1): moving quickly
    - Fast(2): performing some act quickly
    - Fast(3) doing something requiring a short space of time
    - Fast(4): involving rapid motion
    - Etc, etc.
  - These meanings are not listed in the lexicon, but generated by context:
    - Fast typist
    - Fast game
    - Fast book
    - Fast ball

Outline

1. Methods in Lexical Semantics
   (Section 2 of [Pustejovsky91])
2. Existing Representations
   (Sections 3, 4)

3. Proposal: Generative Lexicon
   (Section 5)
4. Details: Qualia Structure & Lexical Inheritance
   (Sections 6, 7)

Methods in lexical semantics? (1/2)

- Identify differences between grammatical classes
  - Verbs are typically predicators, nouns are arguments
- Find distinctions between words of the same class base on collocation and cooccurence tests
  - dog & book differ due to animacy: “dog ran” vs. *“book ran”
- Test for alternations realized in syntax, e.g.
  - (1a) The glass broke. (1b) John broke the glass.
  - (2a) *The bread cut. (2b) John cut the bread.
- Test for context-free vs. context-sensitive entailment
  - (3a) John killed Bill. (3b) Bill died. [killing always entails dying]
  - (5a) Mary finished the cigarette. (5b) Mary finished her beer. [5a entails “finish smoking”, while 5b entails “finish drinking]

Note: The example numbering in these slides—e.g. (5a)—exactly follow those in [Pustejovsky91]

Methods in lexical semantics (2/2)

- Test for word ambiguity.
  - "Accidental" ambiguity (homonym): river bank vs. rich bank
  - "Logical" polysemy:
    - (7a) The bank raised its interest rates yesterday *(i.e. the institution)*
    - (7b) The store is next to the new bank *(i.e. the building)*
- Investigate compositional nature of a word when applied to other words
  - (8a) the alleged suspect. (8b) the female suspect.

    [intension vs. attribute]
  - (8c) An occasional sailor walks by on the weekend.

    [frequency adjective modifying a phrase]
  - (9a) fast typist: one who types quickly. (9b) fast car: one which can move quickly. (9c) fast waltz: one with a fast tempo.

    [interpretation of "fast" depends on the noun it is modifying]

Existing Representations

- Focus on verbs & their classification (e.g. MIT Lexicon Project)

- E.g. Distinguish word senses:
  - (15a) John baked the potato. → change-of-state sense
  - (15b) John baked the cake. → create sense
  - (18a) Mary ran to the store yesterday. → go-by-means-of-running
  - (18b) Mary ran yesterday. → move by running

Outline

1. Methods in Lexical Semantics
   (Section 2 of [Pustejovsky91])
2. Existing Representations
   (Sections 3, 4)

3. Proposal: Generative Lexicon
   (Section 5)
4. Details: Qualia Structure & Lexical Inheritance
   (Sections 6, 7)

Motivating example for GL

(i.e. why is it insufficient to exhaustively list word senses)

- How many senses for the word “closed”?
  - (21a) The door is closed. → state of not open
  - (21b) The door closed. → transition from open to close
  - (22c) John closed the door. → cause to close

- Proposal:
  - A minimal decomposition of “closed” should have an opposition of terms: closed and not-closed.
  - Combined with event structure, Example (21b) and (21c) makes more explicit the transition (i.e. additional meaning is generated)
  - But this is left implicit in (21a).

Four levels of representation in GL

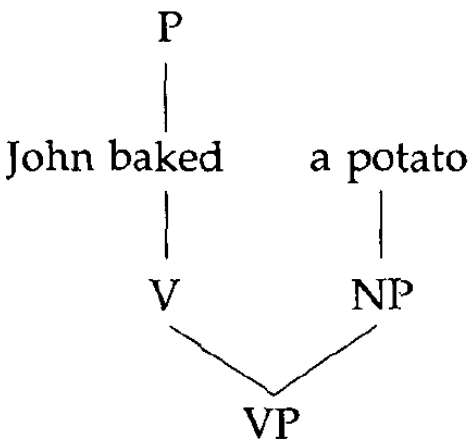
- Argument Structure:
  - Behavior of word as a function and how it maps to syntax
- Event Structure
  - Identifies the event type of a word or phrase, e.g. as state (e^S), process (e^P), or transition (e^T).
- Qualia Structure:
  - The essential attributes of an object as defined by the word
- Inheritance Structure
  - How the word is related to other concepts in the lexicon

Co-compositionality

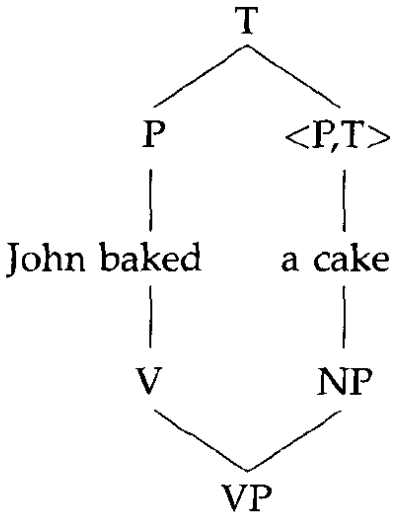
- Enumerating senses for the verb “bake”:
  - (22a) John baked the potato. [$bake_1$ = change(x,State(y))]
  - (22b) John baked the cake. [$bake_2$ = create(x,y)]
- But suppose “bake” only has one change-of-state process verb reading: λ y λ x λ e^P [bake(e^P) $\wedge$ agent(e^P,x) $\wedge$ object(e^P,y)]
- Qualia of potato/cake: *knowledge of object implies not only identification, but also explanation of how it comes about*
  - Potato: natural object
  - Cake: artificial object → sense of create is part of the meaning of cake by virtue of it being an artificial object.
- Thus, the change in meaning of “bake” from (22a) to (22b) comes not from the word itself, but rather in composition with the complement, at the level of the entire verb phrase

Example of verb “bake”

Example 30
a. **bake** as *Process*:
$\exists e^P[bake(e^P) \wedge agent(e^P, j) \wedge object(e^P, a\text{-}potato)]$
b.



Example 31
a. **bake** as a derived *Transition*:[15]
$\exists e^P, e^S[create(e^P, e^S) \wedge bake(e^P) \wedge agent(e^P, j) \wedge object(e^P, x) \wedge cake(e^S) \wedge object(e^S, x)]$
b.



Qualia Structure

- Constitutive Role:
  - The relation between it and its constituent parts, e.g. material, weight, parts
- Formal Role:
  - That which distinguishes it within a larger domain (physical characteristics), e.g. orientation, magnitude, shape, color
- Telic Role:
  - Its purpose and function
- Agentive Role:
  - Factors involved in its origin or "bringing about"

```
novel(*x*)
  Const: narrative(*x*)
  Form: book(*x*), disk(*x*)
  Telic: read(T,y,*x*)
  Agentive: artifact(*x*), write(T,z,*x*)
```

```
dictionary(*x*)
  Const: alphabetized-listing(*x*)
  Form: book(*x*), disk(*x*)
  Telic:  reference(P,y,*x*)
  Agentive: artifact(*x*), compile(T,z,*x*)
```

Metonymy and Type Coercion

- Metonymy: subpart or related object "stands for" the object itself
  - (33a) Mary enjoyed the book. → enjoyed reading the book?
  - (33b) Thatecher vetoed the channel tunnel. → "veto" usually selects for an object that is a legislative bill
  - (33c) John began a novel. → complement is usually a property or action, e.g. [reading/writing a novel]
- Type Coercion in GL:
  - A semantic operation that converts an argument to the type that is expected by a function.
  - Qualia structure of "book" specifies what the artifact is used for, i.e. reading.

Example of type coercion

(33c) John began a novel

1. "begin" requires an argument of type transition

Example 45

$\lambda P_T \lambda \mathcal{P} \mathcal{P} \lambda x[begin'(P_T(x^*))(x^*)]$

2. Qualia structure of the complement "novel":

Example 46

novel translates into:

$\lambda x[novel(x) \wedge Const(x) = narrative'(x) \wedge$
$Form(x) = book'(x) \wedge$
$Telic(x) = \lambda y, e^T[read'(x)(y)(e^T)] \wedge$
$Agent(x) = \lambda y, e^T[write'(x)(y)(e^T)]]$

a. $Q_T(\textbf{novel}) = \lambda y, e^T[read(x)(y)(e^T)]$

b. $Q_A(\textbf{novel}) = \lambda y, e^T[write(x)(y)(e^T)]$

3. By type coercion, we get read/write meaning (both ok)
4. But when the complement has no such interpretation in its qualia, result is odd… (51a) *Mary began a rock.

Example of Adjective-Noun

- Modifiers can apply only to a subset of qualia for a noun, rather than serve as attribute for entire noun
- Example: “fast” applies to telic role

Example 52

a. **a fast car**: driving

$$Q_T(car) = \lambda x \lambda y \lambda e^P [\, drive(x)(y)(e^P) \,]$$

b. **a fast typist**: typing

$$Q_T(typist) = \lambda x \lambda e^P [\, type(x)(e^P) \,]$$

c. **a fast motorway**: traveling

$$Q_T(motorway) = \lambda x \lambda e^P [\, travel(cars)(e^P) \ \wedge on(x)(cars)(e^P) \,]$$

Example of "Figure-Ground" nominals

- Nominals like "window", "door" have 2 interpretations
  - (54a) John crawled through the window. → the opening (aperture)
  - (54b) The window is closed. → the object
  - (55a) Mary painted the door. → the object
  - (55b) Mary walked through the door. → the opening

- "Paint" in (55a) applies to Formal role, while "through" in (55b) applies to Constitutive role.
- Again, meaning of "door" is not enumerated but generated by co-composition

```
door(*x*,*y*)
    Const: aperture(*y*)
    Form: phys-obj(*x*)
    Telic: pass-through(T,z,*y*)
    Agentive: artifact(*x*)
```

Lexical Inheritance

- Fixed inheritance:
  - Static network of relations, e.g. hyponyms, hypernyms
- Projective inheritance:
  - Operates generatively from the qualia structure to create relations for ad hoc categories
- Both are necessary for semantic interpretation of sentences

Projective Inheritance

- Both are well-formed, but (58a) seems more prototypical.
  - (58a) The prisoner escaped last night.
  - (59a) The prisoner ate dinner last night.
- We associate "prisoner" more with "escape". This information comes from projective (not fixed) inheritance.
- Basic procedure:
  - Allow transformations on the value of qualia roles
  - A sequence of transformations generates a "conclusion" space

```
prisoner(*x*)
  Form:  human(*x*)
  Telic: [confine(y,*x*) & location(*x*,prison)]
```

Lexical Semantics for *escape*:

$$\lambda x \lambda e^T \exists e^P, e^S [escape(e^T) \wedge act(e^P) \wedge confined(e^P) \wedge agent(e^P, x) \wedge \neg confined(e^S) \wedge object(e^S, x)]$$

Summary

1. GL is a theory of lexical semantics that spreads the "semantic load" across all words of the sentence.

2. GL does not enumerate senses, but allows meaning to be generated by context → elegantly captures creative usage
   - generative devices include: co-composition, type coercion

3. This is done using a 4-level representation of lexical items:
   - argument structure, event structure, qualia structure, lexical inheritance