Computational Semantics

Generalized quantifiers and scope ambiguity

Chapter 18 [J&M'09]

How to represent the meaning of other quantifiers?

(1) a. Most diplomats left the building.
b. Few diplomats left the building.
c. Only diplomats left the building.
d. Many diplomats diplomats left the building.
e. Six diplomats left the building.
f. Half of the diplomats left the building.
g. More than half of the diplomats left the building.

Quantifiers

Most = more than half

$$_g[\![most_x(\phi,\psi)]\!]^{\mathcal{M}}_g \quad \textit{iff}$$
$$A = \{a : g[x]g' \wedge g'(x) = a \wedge {}_g[\![\phi]\!]^{\mathcal{M}}_{g'}\ \}$$
$$B = \{b : g[x]g'' \wedge g''(x) = b \wedge {}_g[\![\psi]\!]^{\mathcal{M}}_{g''}\ \}$$
$$|A \cap B| > |A \backslash B|$$

(where '$\backslash$' is set difference)

Example:

Suppose we have $\mathcal{M} = \langle D, F\rangle$ as

- $D = \{a, b, c, d, e, f, g\}$
- $F{=}\{\langle cat, \{a, b, c, d, e\}\rangle, \langle dog, \{f, g\}\rangle, \langle sneeze, \{a, b, c, f, g\}\rangle\}$

Then, $most_x(cat(x), sneeze(x))$ yields:

$A = \{a, b, c, d, e\}$ $\qquad$ $B = \{a, b, c, f, g\}$,

$|A \cap B| = \{a, b, c\}$ $\qquad$ $|A\backslash B| = \{d, e\}$

Quantifiers

Many

${}_g[\![many_x(\phi, \psi)]\!]_g^{\mathcal{M}}$ *iff*

$$A = \{a : g[x]g' \wedge g'(x) = a \wedge {}_g[\![\phi]\!]_{g'}^{\mathcal{M}} \}$$
$$B = \{b : g[x]g'' \wedge g''(x) = b \wedge {}_g[\![\psi]\!]_{g''}^{\mathcal{M}} \}$$

$|A \cap B| > m$ (cardinal interpretation) or

$|A \cap B|/|A| > m$ (proportional interpretation)

Example:

- Cardinal:

  (2) a. Many shots were fired.

  b. Lee found many books in a closet.

- Proportional:

  (3) a. Many stars are red.

  b. Many fleas were tested. Few survived.

Quantifiers

six

$$
\begin{aligned}
& {}_{g}[\![six_x(\phi,\psi)]\!]_g^{\mathcal{M}} \quad \textit{iff} \\
& \qquad A = \{a : g[x]g' \wedge g'(x) = a \wedge {}_{g}[\![\phi]\!]_{g'}^{\mathcal{M}} \} \\
& \qquad B = \{b : g[x]g'' \wedge g''(x) = b \wedge {}_{g}[\![\psi]\!]_{g''}^{\mathcal{M}} \} \\
& \qquad |A \cap B| = 6
\end{aligned}
$$

Numerals, more generally:

$$
\begin{aligned}
& {}_{g}[\![N_x(\phi,\psi)]\!]_g^{\mathcal{M}} \quad \textit{iff} \\
& \qquad A = \{a : g[x]g' \wedge g'(x) = a \wedge {}_{g}[\![\phi]\!]_{g'}^{\mathcal{M}} \} \\
& \qquad B = \{b : g[x]g'' \wedge g''(x) = b \wedge {}_{g}[\![\psi]\!]_{g''}^{\mathcal{M}} \} \\
& \qquad |A \cap B| = N
\end{aligned}
$$

(note: often it is assumed that $|A \cap B| \geqslant N$)

no / zero

$$
{}_g[\![no_x(\phi,\psi)]\!]_g^{\mathcal{M}} \quad \textit{iff}
$$

$$
\begin{aligned}
&A = \{a : g[x]g' \wedge g'(x) = a \wedge {}_g[\![\phi]\!]_{g'}^{\mathcal{M}}\ \}\\
&B = \{b : g[x]g'' \wedge g''(x) = b \wedge {}_g[\![\psi]\!]_{g''}^{\mathcal{M}}\ \}\\
&|A \cap B| = \varnothing
\end{aligned}
$$

half of N

$$
{}_g[\![\textit{half-of}_x(\phi,\psi)]\!]_g^{\mathcal{M}} \quad \textit{iff}
$$

$$
\begin{aligned}
&A = \{a : g[x]g' \wedge g'(x) = a \wedge {}_g[\![\phi]\!]_{g'}^{\mathcal{M}}\ \}\\
&B = \{b : g[x]g'' \wedge g''(x) = b \wedge {}_g[\![\psi]\!]_{g''}^{\mathcal{M}}\ \}\\
&|A \cap B| = |A|/2
\end{aligned}
$$

Quantifiers

We can use the same approach to represent $\exists$ and $\forall$:

all / **every** / **each**

$$_g[\![\forall_x(\phi,\psi)]\!]_g^{\mathcal{M}} \quad \textit{iff}$$
$$A = \{a : g[x]g' \wedge g'(x) = a \wedge {}_g[\![\phi]\!]_{g'}^{\mathcal{M}} \}$$
$$B = \{b : g[x]g'' \wedge g''(x) = b \wedge {}_g[\![\psi]\!]_{g''}^{\mathcal{M}} \}$$
$$A \subseteq B$$

some / **a(n)**

$$_g[\![\exists_x(\phi,\psi)]\!]_g^{\mathcal{M}} \quad \textit{iff}$$
$$A = \{a : g[x]g' \wedge g'(x) = a \wedge {}_g[\![\phi]\!]_{g'}^{\mathcal{M}} \}$$
$$B = \{b : g[x]g'' \wedge g''(x) = b \wedge {}_g[\![\psi]\!]_{g''}^{\mathcal{M}} \}$$
$$A \cap B \neq \emptyset$$

Quantifiers interact is complex ways:

(4) Every dog chased a cat. (ambiguous)
$\forall x(dog(x) \Rightarrow \exists y(cat(y) \wedge chase(x,y)))$
$\exists y(cat(y) \wedge \forall x(dog(x) \Rightarrow chase(x,y)))$

(5) Many dogs chased a cat. (ambiguous)
$Many_x(dog(x) \Rightarrow \exists y(cat(y) \wedge chase(x,y)))$
$\exists y(cat(y) \wedge Many_x(dog(x) \Rightarrow chase(x,y)))$

Quantifier scope ambiguity is NOT lexical ambiguity:

(6) a. Every dog chased every cat.
$\forall x(dog(x) \Rightarrow \forall y(cat \Rightarrow chase(x,y)))$
$\forall y(cat(y) \Rightarrow \forall x(dog(x) \Rightarrow chase(x,y)))$ (redundant)

b. A dog chased a cat.
$\exists x(dog(x) \wedge \exists y(cat \wedge chase(x,y)))$
$\exists y(cat(y) \wedge \exists x(dog(x) \wedge chase(x,y)))$ (redundant)

More examples of semantic scope ambiguity:

(7) a. Every student speaks two languages.
b. An aide accompanied every guest to the correct table.
c. Tom didn't read a book.
d. Someone always wins this game.
e. John will force you to marry no one.
f. John does not speak exactly three languages.
g. Kim is looking for a parrot that can talk.

World knowledge often interferes with scope resolution:

(8) Every hunter killed a duck. (different ducks)

Continuations resolve scope, sometimes creating humor:

(9) Every fifteen seconds a man is arrested in the US. Poor guy!

Combinatorial explosion of interpretations

(10) Every dog chased two cats up a tree.

$\forall x(dog(x) \Rightarrow \exists y(tree(y) \wedge \exists w(cats(w) \wedge |w| = 2 \wedge chase(x, w, y))))$
$\forall x(dog(x) \Rightarrow \exists w(cats(w) \wedge |w| = 2 \wedge \exists y(tree(y) \wedge chase(x, w, y))))$
(redundant)
$\exists y(tree(y) \wedge \forall x(dog(x) \Rightarrow \exists w(cats(w) \wedge |w| = 2 \wedge chase(x, w, y))))$
$\exists w(cats(w) \wedge |w| = 2 \wedge \forall x(dog(x) \Rightarrow \exists y(tree(y) \wedge chase(x, w, y))))$
$\exists w(cats(w) \wedge |w| = 2 \wedge \exists y(tree(y) \wedge \forall x(dog(x) \Rightarrow chase(x, w, y))))$
$\exists y(tree(y) \wedge \exists w(cats(w) \wedge |w| = 2 \wedge \forall x(dog(x) \Rightarrow chase(x, w, y))))$
(redundant)

$3! = 6$ combinations, 4 non-equivalent.

Quantifiers

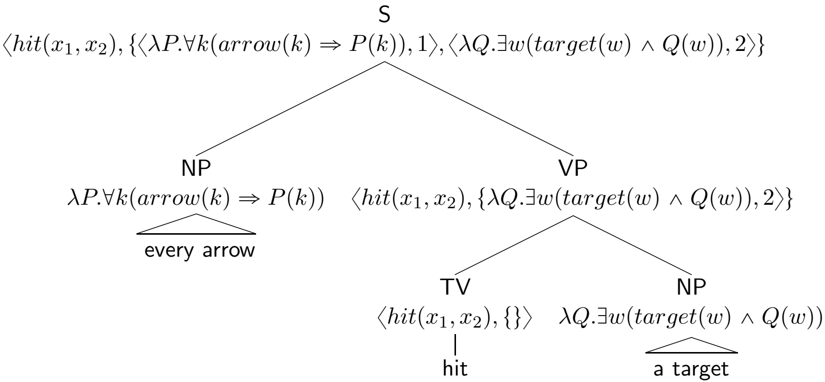
The number of combinations is at most $n!$ and at least C_n

quantifiers (n)	factorial combinations ($n!$)	catalan number C_n
2	2	2
3	6	5
4	24	14
5	120	42
6	720	132

Solution to this combinatorial explosion problem:
Don't resolve scope. In applications like machine translation, scope ambiguities do not matter.

- Quantifier Storage (Cooper Storage; Keller Storage);
- Express semantic structure with a constraint-based system, where scope is resolved dynamically.
  Section 18.3 of [J&M'00]

Cooper Storage

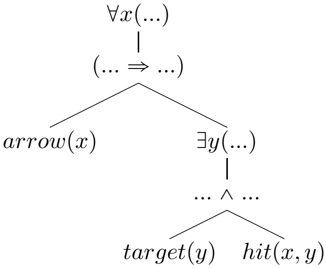


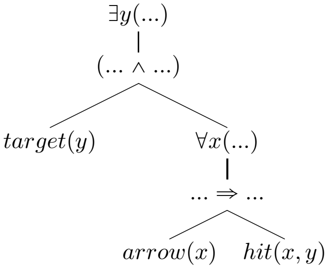
Retrieval:

If $\langle \psi, \{(\phi, i)\} \cup Y\rangle$ then $\langle (\phi)(\lambda x_i.\psi), Y\rangle$ such that x_i occurs in ψ

Constraint-based scope underspecification

Formulas are trees:

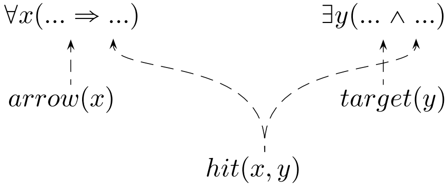




Patterns:

- The variable bindings are the same;
- The N semantics ends up in the same place
- The V semantics ends up in the deepest part of the formula

The graph below captures these patterns in a single structure:

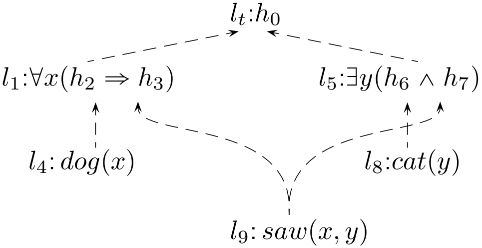


The arrows mean '*put this here (directly or indirectly)*'

We can view these structures as graphs:

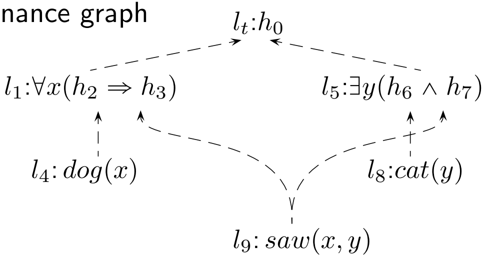
- Each FOL expression is labeled (l_n);
- FOL arguments are 'holes' (h_m);
- Semantic structure is expressed by dominance constraints between labels and holes.

This way, **multiple scopes** can be represented by **a semi-lattice**:



Quantifiers

The dominance graph



Is equivalent to:

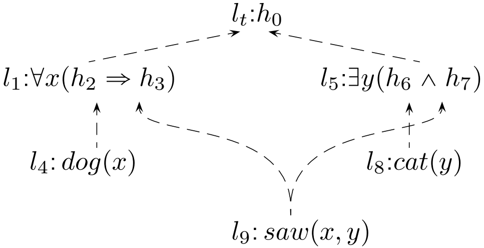
$$L = \{l_t : h_0, l_1 : \forall x(h_2 \Rightarrow h_3),\ l_4 : dog(x),\ l_5 : \exists y(h_6 \wedge h_7),\ l_8 : cat(y),\ l_9 : saw(x, y)\}$$
$$C = \{l_1 \leqslant h_0, l_5 \leqslant h_0, l_9 \leqslant h_0, l_4 \leqslant h_2, l_8 \leqslant h_6, l_9 \leqslant h_3, l_9 \leqslant h_7\}$$

Where:

- '$\leqslant$' means 'equal or dominated by'.
- '$\leqslant$' constraint can be resolved as '$=$' or as '$<$'.

Quantifiers

A disambiguation of an underspecified representation is a bijection from h to l where the resulting graph is a tree.



Two possible disambiguations:

(11) a. $l_1 = h_0, l_4 = h_2, l_5 = h_3, l_8 = h_6, l_9 = h_7$

b. $l_5 = h_0, l_8 = h_6, l_1 = h_7, l_4 = h_2, l_9 = h_3$

No other disambiguation obeys the $\leqslant$ constraints and yields a tree.

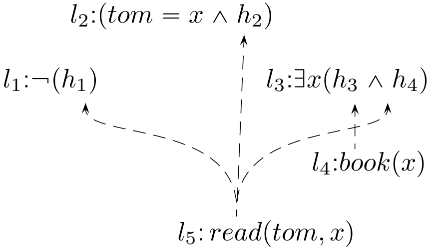
- S_{l_5} $\langle\{l_1{:}\forall x(h_2 \Rightarrow h_3),\ l_4{:}dog(x),\ l_5{:}bark(x)\},\ \{l_4 \leqslant h_2, l_5 \leqslant h_3\}\rangle$
  - NP_{h_2,h_3} $\langle\{l_1{:}\forall x(h_2 \Rightarrow h_3),\ l_4{:}dog(x)\},\ \{l_4 \leqslant h_2\}\rangle$
    - D_{h_2,h_3} $\langle\{l_1{:}\forall x(h_2 \Rightarrow h_3)\},\ \{\}\rangle$
      - *every*
    - N_{l_4} $\langle\{l_4{:}dog(x)\},\ \{\}\rangle$
      - *dog*
  - VP_{l_5} $\langle\{l_5{:}bark(x)\},\ \{\}\rangle$
    - *barked*

The same idea applies to other scope ambiguities:

(12) a. Tom didn't read a book.

b. $\exists y(book(y) \wedge \neg read(tom, y))$

c. $\neg(\exists y(book(y) \wedge read(tom, y)))$



Some online resources about constraint-based underspecification

- Semantic underspecification in the broader context of CL: here
- Detailed theoretical discussion – and Prolog implementation of a $\lambda-$calculus based grammar & scope resolution – found here

Corpora with FOL semantics

- CCGBank, parsed PennTreebank.
- Paralel Meaning Bank (English, German, Dutch and Italian)
- You can create your own corpora, using the OpenCCG Java suite (see also EasyCCG parser)

Machine learning of λ-term representations

- Wong & Mooney (2007) use a CFG to learn lexical semantic representations from a training corpus.
  Link to ACL paper here
- Zettlemoyer & Collins (2005) use probabilistic CCG grammars to model the combination of lambda terms directly.
- See also Liang et al., 2011; Berant et al., 2013; Kwiatkowski et al., 2013; Artzi and Zettlemoyer, 2013; Kushman and Barzilay, 2013, and Wang et al. 2015.

SOTA accuracy of 89% exact matches on the Geo880 corpus and 82% on the ATIS speech corpus (see here for an overview)