# Semantic Theory

## Lexical Semantics 2

Manfred Pinkal/ Stefan Thater
Saarland University
Summer 2009

## What is role information good for?

- Providing a level of representation for the mapping between syntactic complements and semantic argument positions (role-linking).
- Encoding semantic correspondences between different relational expressions and different uses of one relational expression in a systematic way.
- What is the place of role information in a linguistic framework?
  - Grammar or Lexicon?
  - Part of semantic representation, or only information used in the syntax-semantics interface?



## Grammar or Lexicon?

- Role information is part of the grammar: Role linking regularities can be systematically described. Example:
  - The agent takes the subject position.
  - In the absence of an agent, the instrument takes the subject position.
  - In the absence of agent and instrument, the theme takes the subject position.
- Role information is part of the lexicon: Role linking information is stated explicitly for each lemma. Example:

  *give:* SB → *Agent*
  *OA* → *Theme*
  *OD* → *Recipient*
- Providing grammatical theories for role linking has been a challenge for theoretical linguists. For practical purposes of correct wide-coverage grammar engineering the lexicon is the more useful alternative.



## Are roles part of the semantic representation?

If roles are not part of the semantic representation, then ...

- Semantic representations are simple FOL predicate-argument structures.
- But: Frame and role information is a highly productive source for systematic inference.
- Example: *If a giving event occurs at time t, with agent a, recipient b, and theme c, then a is a person/ animate individual who had b at a time t'<t, b is a person/ animate individual who has b at a time t''>t.*
- *Therefore: Role information* should not be thrown away, but kept in the semantic representation. But: How?

# Event Semantics

Interpretation of sentences containing adjuncts:

*(1) The gardener killed the baron at midnight in the park*

$\Rightarrow kill_4(g, b, m, p)$

*(2) The gardener killed the baron at midnight*

$\Rightarrow kill_3(g, b, m)$

*(3) The gardener killed the baron in the park*

$\Rightarrow kill_2(g, b, p)$

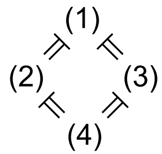
*(4) The gardener killed the baron*

$\Rightarrow kill_1(g, b)$



# Davidson's Problem

- Problem: How can the logical entailment relations between the different uses of *kill* be systematically explained?




- Naïve FOL interpretation does not solve the problem:
  - $kill_4(g, b, m, p) \not\models kill_3(g, b, m)$
  - $kill_3(g, b, m) \not\models kill_1(g, b)$
  - etc.



# Adjunct Interpretation: Second Attempt

- Determine the maximum arity n of the predicate.
- Take n to be the arity of the predicate.
- Bind syntactically empty argument positions with existential quantifier.

  (1) $\Rightarrow kill(g, b, m, p)$

  (2) $\Rightarrow \exists y\, kill(g, b, m, y)$

  (3) $\Rightarrow \exists x\, kill(g, b, x, p)$

  (4) $\Rightarrow \exists x \exists y\, kill(g, b, x, y)$

- Problem: What is the maximum arity of a predicate?
  *The gardener killed the baron at midnight in the park under cover of absolute darkness with a shotgun …*



# Davidson's Proposal

- Standard FOL-Semantics: two-place verbs denote sets of pairs of individuals.
- Davidson: Verbs expressing events have an additional event argument, which is not realised at linguistic surface:

  kill(e,x,y)

- In general, event verbs are represented by relations of a fixed arity (number of obligatory complements +1)
- Adjuncts express two-place relations between events and the respective "cirumstantial information" (a time, a location, ...)
- The event variable is existentially bound:

  *The gardener killed the baron at midnight in the park*

  $\Rightarrow \exists e[\, kill(e,g,b) \wedge time(e, m) \wedge location(e, p)\, ]$

## The solution to Davidson's problem

- Event semantics admits an arbitrary, open number of adjuncts.
- Since adjunct information is added through conjunction, entailment between a sentence with adjunct and the sentence without adjunct is trivial:

  $\exists e[\ kill(e,g,b) \wedge time(e,\ m) \wedge location(e,\ p)\ ]$

  $\models \exists e[\ kill(e,g,b) \wedge time(e,\ m)\ ]$

  $\models \exists e[\ kill(e,g,b)\ ]$



## Representation of Thematic Roles

- Treat complements analogously to adjuncts:
- Thematic roles are two-place relations between the event denoted by the verb, and an argument role filler.
- The event verb itself is just a one-place predicate taking an event as argument.
- Example:

  *The gardener killed the baron at midnight in the park*

  $\Rightarrow \exists e\ [kill(e) \wedge ag(e,g) \wedge pat(e,b) \wedge time(e,m) \wedge location(e,p)]$

  or, using FrameNet frames and roles:

  $\exists e\ [killing(e) \wedge killer(e,g) \wedge victim(e,b)]$
- This analysis is called „Neo-Davidsonian“ semantics.
- It allows to partition semantic information into minimal pieces pieces of information: One-place predicates and two-place relations.



## Model-theoretic interpretation

- Assume two disjoint classes, or kinds, or sorts of individuals:
  - A set of "standard individuals" or "objects" $U_O$
  - A set of events $U_E$
- A model structure is defined as

  $M = \langle\langle U_O, U_E \rangle, V\rangle$,

  with $U_O \cap U_E = \emptyset$,

  V interpretation function like in standard FOL, with domain $U = U_O \cup U_E$



## Variables in sorted logic

- We assume a separate inventory of variables for each sort of individuals:
  - (Standard) Object variables: $Var_O$ = x, y, z, ... a , $x_1$, $x_2$, ...
  - Event variables: $Var_E$ = e, e', e'', ..., , $e_1$, $e_2$, ...
- Variable assignment functions g assign object and event variables individuals of the respective sort-specific domain:
  - $g(x) \in U_O$ for $x \in Var_O$
  - $g(e) \in U_E$ for $x \in Var_E$
- Quantification ranges over sort-specific domains:
  - $[[\exists x\ \Phi\ ]]^{M,g} = 1$ iff there is an $a \in U_O$ s.t. $[[\ \Phi\ ]]^{M,g[x/a]} = 1$
  - $[[\exists e\ \Phi\ ]]^{M,g} = 1$ iff there is an $a \in U_E$ s.t. $[[\ \Phi\ ]]^{M,g[x/a]} = 1$

## Variables in sorted logic

- Quantification with sorted variables is an elegant alternative to standard FOL notation with explicit constraints.
  (1) "The agent of every giving event is an animate entity"
  (2) $\forall e \forall x$ (giving(e) $\wedge$ agent(e,x) $\rightarrow$ animate(x))
  (3) $\forall y$(Event(x) $\rightarrow$ $\forall x$ (Object(x) $\rightarrow$ (giving(y) $\wedge$ agent(y,x) $\rightarrow$ animate(x))))
- Sorted logic representation (2) is equivalent with standard FOL (3), given that "Event" and "Object" are predicates with the set of all standard objects and the set of all events as fixed denotations.



## Events as individuals

- Events as individuals: Technical trick or appropriate extension of semantics?
- Conceiving of a coughing (killing, selling, ...) as an entity in its own right is intuitive:
- Interpreting the predicate *cough* as a class of events with a certain property is much more natural than defining it as the set of all coughing persons (in a givn world and time).
- Also, it provides a natural basis for the interpretation of (deverbal or genuine) event nouns*: the cough, the murder, the purchase, the sale* refer to event individuals, like *student, tree, table* refer to standard individuals.



## Events as individuals

- But: Events seem to be more problematic individuals than students, trees, tables:
  (1) Events are typically complex, without clear criteria for delimitation and identity.
  (2) Events are typically volatile, are temporally limited and temporally ordered.
- While (2), the narrow connection with time, is really event-specific, a closer inspection shows that the problems in (1) also occur with standard objects, denoted by standard nouns/ noun phrases.
- We will next look into some phenomena of nominal semantics, and come back to verbs and events afterwords.



## Plural Noun Phrases

*Bill and Mary work* ⊨ *Bill works*

*Bill and Mary work* ⊨ *Mary works*

work(b) $\wedge$ work(m) ⊨ work(b)

work(b) $\wedge$ work(m) ⊨ work(m)

*The students work , John is one of the students*

⊨ *John works*

$\forall$x(student(x) $\rightarrow$ work(x)), student(j) ⊨ work(j)

# Collective predicates

*Bill and Mary met* $\not\models$ *Bill met*

*The students met , John is one of the students*
$\not\models$ *John met*

*The committee met, John is member of the committee*
$\not\models$ *John met*

- "meet" is a collective predicate.



# Collective predicates

- Collective predicates are only applicable with plural or group NPs. Their semantics cannot be reduced to atomic statements about single standard individuals.
- Examples for collective predicates:
  - *meet, gather, unite, agree, be similar, disperse, disagree, be numerous, ...*
- Distributive or individual level predicates like *work, sleep, eat, tall* apply to singular and plural nouns. A predication with a plural NP "distributes" over the individual objects covered by the NP.

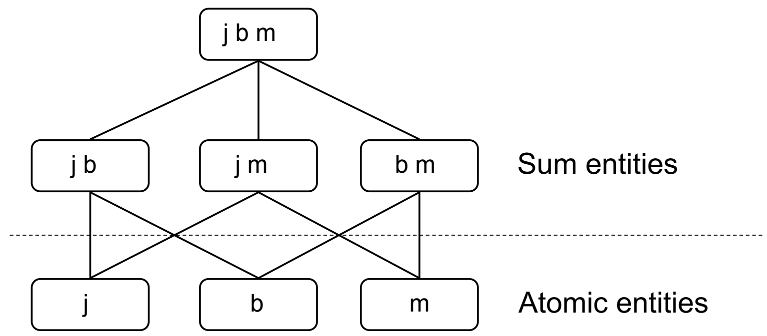


# Interpretation of Plural Terms

- In addition to standard individuals, we add "group" or "sum" entities to the model structure universe.
- Singular expressions denote standard "atomic" entities, plural and group expressions denote sums.
- To model the entailment relations between the group and its members, e.g., in the context of distributive predicates, we also add the membership or "individual part" relation to the model structure.



# Structured Universe – Example



## Lattices and Semi-lattices

- A partial order is a structure <A, ≤> with reflexive, transitive, and antisymmetric ≤ .
- Let <A, ≤> be a partial order:

  The join of a and b ∈ A: $a \sqcup b$ is the lowest upper bound for a and b.

  The meet of a and b ∈ A: $a \sqcap b$ is the highest lower bound for a and b.
- A lattice is a partial order <A, ≤> which is closed under meet and join.



## Lattices and Semi-lattices

- A lattice may or may not have one maximal and minimal element. If it has such elements, they are named **1** and **0**, respectively, and the lattice is called bounded.
- An a ∈ A is an atom, if a ≠ **0** and there is no b ≠ **0** in A such that b<a.
- A lattice <A, ≤> is atomic, if for every a ≠ **0 there is an atom b ≤ a.**
- A join semi-lattice is a partial order <A, ≤> which is closed under join.



## Model structure for plural terms

- A model structure is a pair M = ⟨⟨U, ≤⟩, V⟩, where
  - <U ,≤> is an atomic join semi-lattice with universe U and individual part relation ≤.
  - V is a value assignment function.
- A⊆U is the set of atoms in <U ,≤>.
- U\A is the set of non-atomic elements, i.e., the sum objects in U.



## Logic for plural terms

- Like standard FOL. We add a summation operator ⊕, a one-place predicate At for "atom" and a two-place relation ◁ for "(proper) individual part".
- Application examples:
  - j ⊕ b, translating "John and Bill"
  - j ⊕ b ◁ the_committee: "John and Bill are individual parts / members of the committee"

# Interpretation

Like standard FOL interpretation

- with additional clauses for $\oplus$ and $\triangleleft$ :

  $[\![a \oplus b]\!]^{M,g} = [\![a]\!]^{M,g} \sqcup [\![b]\!]^{M,g}$

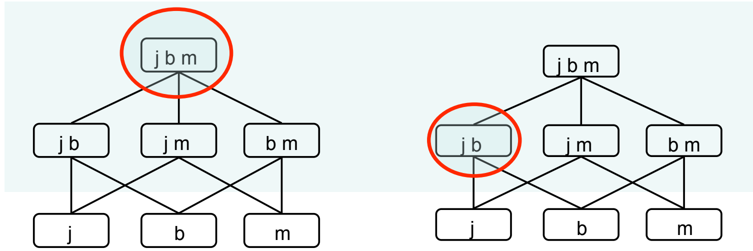
  $[\![a \triangleleft b]\!]^{M,g} = 1$ iff $[\![a]\!]^{M,g} < [\![b]\!]^{M,g}$

  $[\![At(a)]\!]^{M,g} = 1$ iff $[\![a]\!]^{M,g} \in A$
- with standard individual constants (proper names, definite descriptions) denoting atoms ($V_M(a) \in A$),
- group constants (plural and group NPs) denoting sums ($V_M(a) \in U \backslash A$)
- predicates satisfying specific constraints with respect to their collectivity status



# Collective predicates

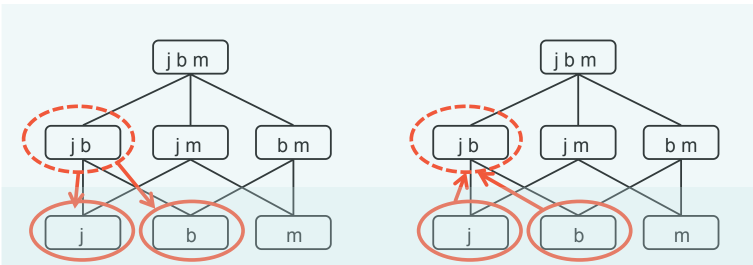
- Collective predicates F (like *meet, collaborate*):
  $V_M(F) \subseteq U \backslash A$






# Distributive predicates

- Distributive predicates F (like *work, tall, student*):
  $V_M(F) \subseteq U$, such that the following axiom is satisfied:
  $\forall x(F(x) \leftrightarrow \forall y( At(y) \wedge y \triangleleft x \rightarrow F(y)))$




$\rightarrow$: Distributivity $\leftarrow$ : Closure under summation

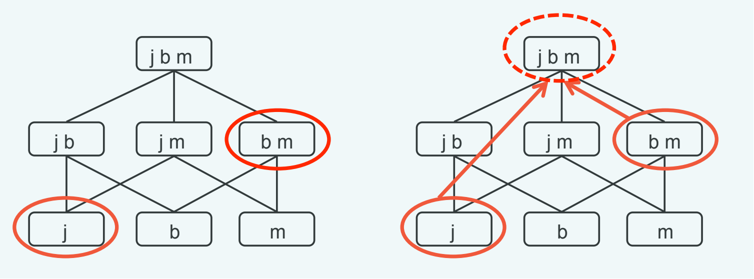


# Distributive predicates

- The denotation of distributive predicates is uniquely determined by their atomic members.
- If a predicate applies to a set $M \subseteq A$, then the full denotation of the predicate is the join semi-lattice generated by M.

## Interpretation of predicates

- Mixed predicates F (e.g., *carrying a piano, solving the exercise*):
  $V_M(F) \subseteq U$




Non-distributive, but closed under summation



## Mass Nouns

- *Examples: water, gold, wood, money, soup, ...*
- Mass nouns don't have individual entities. They are "divisive": Every amount of water can be subdivided into proper parts which are also amounts of water.
- The basic property of mass nouns is a basic problem for FOL-based semantics: There are no atomic individuals as basic material of model structures.
- The close similarity between plurals and mass nouns suggests a solution to this problem.



## Plural nouns and mass nouns

- Plural nouns and mass nouns
  - are both cumulative:
    If you add students to students, the resulting group are students.
    If you add water to water, the resulting entity is an amount of water.
  - come both with cardinalities: 5 students, 5 l of water.
  - share grammatical properties; e.g., indefinite plural and mass term NPs don't take an article.



## Model structure for mass nouns

- We add a sort of material entities M, which forms a join semi-lattice $\langle M, \leq \rangle$ with the material part relation ≤ – similar to the semi-lattice for plurals, except the facts that $\langle M, \leq \rangle$ is non-atomic and dense.
  - An ordering relation ≤ is dense, if the following holds:
    $\forall x \forall y\, (x<y \rightarrow \exists z (x<z \wedge z<y)$
    E.g., the (strict) order of real numbers is dense, the order of natural numbers is not.
- To distinguish the individual part and the material part relation, we write $\leq_i$ for the former, and $\leq_m$ for the latter, and arrive at the following (intermediate) model structure concept:
- $M = \langle\langle U, \leq_i\rangle, \langle M, \leq_m\rangle, V\rangle$
- In the logical language, we add a material summation operation and a material part relation, and distinguish $\oplus_i$, $\oplus_m$, $\triangleleft_i$, and $\triangleleft_m$.

# Model structure for mass nouns

- There is a narrow relation between the two sub-domains of (singular and plural) individuals and material entities: Each individual consists of a portion of matter.
- We take this relation into account by introducing a “materialisation” function *h* into the model structure: a homomorphism that maps (atomic and pluralic) individuals to the matter they consist of.
- $M = \langle\langle U, \leq_i\rangle, \langle M, \leq_m\rangle, h, V\rangle$
- Because h is a homomorphism, the following hold:

  $a \leq_i b$ iff $h(a) \leq_m h(b)$

  $h([\![a \oplus_i b]\!]^{M,g}) = h([\![a]\!]^{M,g}) \oplus_m h([\![b]\!]^{M,g})$

# Examples

We use two sorts of variables:

x, y, z, ... for (atomic and sum) individuals

X, Y, Z, ... for portions of matter

m for the matter function

*The ring is made of gold*

→ $\exists y(\text{ring}(y) \wedge \text{gold}(m(y)))$

*The ring contains gold*

→ $\exists y \exists X\, (\text{ring}(y) \wedge X \triangleleft_m m(y) \wedge \text{gold}(X))$