Semantic Theory
week 7 – Event semantics

Noortje Venhuizen

Universität des Saarlandes

Summer 2017

A problem with verbs and adjuncts

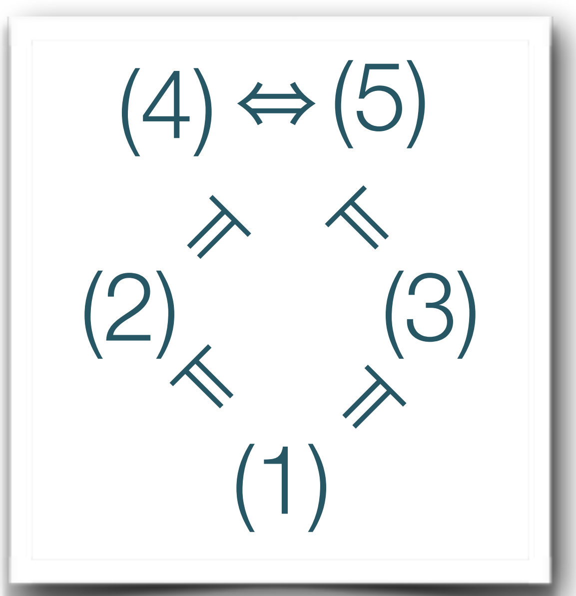
(1) *The gardener killed the baron* $\mapsto$ $kill_1(g',b')$ $kill_1 :: \langle e,\langle e,t\rangle\rangle$

(2) *The gardener killed the baron in the park* $\mapsto$ $kill_2(g',b',p')$ $kill_2 :: \langle e,\langle e,\langle e,t\rangle\rangle$

(3) *The gardener killed the baron at midnight* $\mapsto$ $kill_3(g',b',m')$ $kill_3 :: \langle e,\langle e,\langle e,t\rangle\rangle$

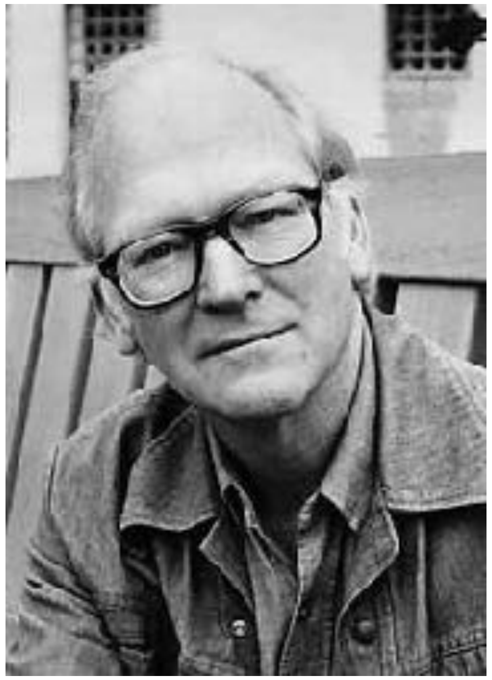
(4) *The gardener killed the baron at midnight in the park* $\mapsto$ $kill_4(g',b',m',p')$ $kill_4 :: \dots$

(5) *The gardener killed the baron in the park at midnight* $\mapsto$ $kill_5(g',b',p',m')$ $kill_5 :: \dots$



Q: How to explain the systematic logical entailment relations between the different uses of “kill”?

Davidson’s solution: verbs introduce events.



Verbs expressing events have an additional event argument, which is not realised at linguistic surface:

- kill ↦ λyλxλe(kill’(e,x,y)) :: ⟨e,⟨e,⟨e,t⟩⟩⟩ *arity = n+1*

Sentences denote sets of events:

- λyλxλe(kill’(e,x,y))(b’)(g’) $\Rightarrow^{\beta}$ λe(kill’(e, g’, b’)) :: ⟨e,t⟩

Existential closure turns sets of events into truth conditions

- λP∃e(P(e)) :: ⟨⟨e,t⟩,t⟩
- λP∃e(P(e))(λe(kill’(e,g’,b’))) $\Rightarrow^{\beta}$ ∃e(kill’(e,g’,b’)) :: t

Davidson (1967, 1980)

Davisonian events and adjuncts

Adjuncts express two-place relations between events and the respective "circumstantial information": time, location, …

- at midnight $\mapsto \lambda P \lambda e(P(e) \wedge \mathtt{time}(e,m'))$ :: $\langle\langle e,t\rangle,\langle e,t\rangle\rangle$
- in the park $\mapsto \lambda P \lambda e(P(e) \wedge \mathtt{location}(e,p'))$ :: $\langle\langle e,t\rangle,\langle e,t\rangle\rangle$

The gardener killed the baron at midnight in the park

$$\left.\begin{array}{l}\mapsto \exists e\,(\text{kill}(e, g', b') \wedge \text{time}(e, m) \wedge \text{location}(e, p')) \\ \Leftrightarrow \exists e\,(\text{kill}(e, g', b') \wedge \text{location}(e, p) \wedge \text{time}(e, m'))\end{array}\right\} \begin{array}{l}\models \exists e\,(\text{kill}(e, g', b') \wedge \text{time}(e, m')) \\ \models \exists e\,(\text{kill}(e, g', b') \wedge \text{location}(e, p')) \\ \models \exists e\,(\text{kill}(e, g', b'))\end{array}$$

Compositional derivation of event-semantic representations

the gardener killed the baron

$\lambda x_e \lambda y_e \lambda e_e[\text{ kill}(e, y, x)](b')(g') \Rightarrow^{\beta} \lambda e [\text{ kill}(e, g', b')]$

… at midnight

α

$\lambda F_{\langle e,t\rangle} \lambda e_e [F(e) \wedge \text{time}(e, m')](\lambda e_1 [\text{ kill}(e_1, g', b')]) \Rightarrow^{\beta} \lambda e [\text{ kill}(e, g, b) \wedge \text{time}(e,m')]$

… in the park

$\lambda F_{\langle e,t\rangle} \lambda e_e [F(e) \wedge \text{location}(e, p')] (\lambda e_2 [\text{kill}(e_2, g', b') \wedge \text{time}(e_2, m')]) \Rightarrow^{\beta}$

$\lambda e [\text{kill}(e, g', b') \wedge \text{time}(e, m') \wedge \text{location}(e, p')]$

Existential closure

$\lambda P_{\langle e,t\rangle} \exists e(P(e))(\lambda e'(K \wedge T \wedge L) \Rightarrow^{\beta} \exists e [\text{ kill}(e, g', b') \wedge \text{time}(e, m') \wedge \text{location}(e, p')]$

Model structures with events

To interpret events, we need enriched *ontological* information

Ontology: The area of philosophy identifying and describing the basic “categories of being” and their relations.

A model structure with events is a triple $M = \langle U, E, V \rangle$, where

- U is a set of “standard individuals” or “objects”
- E is a set of events
- $U \cap E = \varnothing$,
- V is an interpretation function like in first order logic

Sorted (first-order) logic

A variable assignment g assigns individuals (of the correct sort-specific domain) to variables:

- $g(x) \in U$ for $x \in VAR_U$ $VAR_U = \{ x, y, z, \dots , x_1, x_2, \dots \}$ (Object variables)
- $g(e) \in E$ for $e \in VAR_E$ $VAR_E = \{ e, e', e'', \dots, e_1, e_2, \dots \}$ (Event variables)

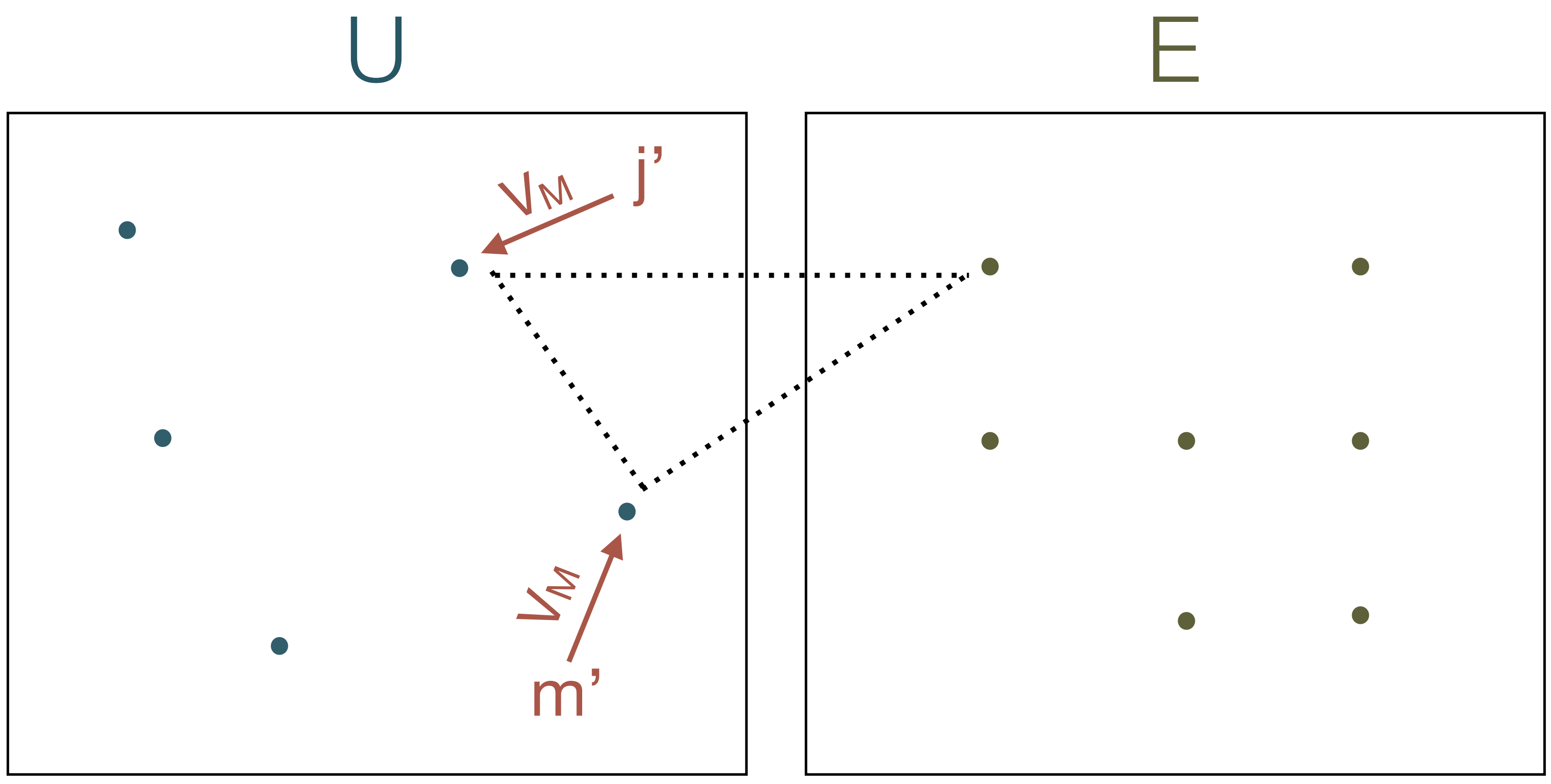
Quantification ranges over sort-specific domains:

- $[\![\exists x\, \Phi]\!]^{M,g} = 1$ iff there is an $a \in U$ such that $[\![\Phi]\!]^{M,g[x/a]} = 1$
- $[\![\exists e\, \Phi]\!]^{M,g} = 1$ iff there is an $a \in E$ such that $[\![\Phi]\!]^{M,g[e/a]} = 1$
- (universal quantification analogous)

Interpreting events

John kisses Mary ↦ ∃e (kiss(e, j’, m’))

⟦ ∃e (kiss(e, j’, m’)) ⟧^{M,g} = 1
iff there is an s ∈ E such that ⟦ kiss(e, j’, m’) ⟧$^{M,g[e/s]}$ = 1
iff there is an s ∈ E such that ⟨s, V_M(j’), V_M(m’)⟩ ∈ V_M(kiss)



Advantages of Davidsonian events

- [x] Intuitive representation and semantic construction for adjuncts
- [ ] Uniform treatment of verb complements
- [ ] Uniform treatment of adjuncts and post-nominal modifiers
- [ ] Coherent treatment of tense information
- [ ] Highly compatible with analysis of semantic roles

Uniform treatment of verb complements

(1) *Bill saw an elephant*

↦ ∃e ∃x (see(e, b', x) ∧ elephant(x))

see :: ⟨e,⟨e,⟨e,t⟩⟩

(2) *Bill saw an accident*

↦ ∃e ∃e' (see(e, b, e') ∧ accident(e'))

see :: ⟨e,⟨e,⟨e,t⟩⟩

(3) *Bill saw the children play*

↦ ∃e ∃e' (see(e, b, e') ∧ play(e', the-children))

see :: ⟨e,⟨e,⟨e,t⟩⟩

Uniform treatment of adjuncts and post-nominal modifiers

Treatment of adjuncts as predicate modifiers, analogous to attributive adjectives:

- red ↦ λF λx [F(x) ∧ red*(x)] ⟨⟨e,t⟩,⟨e,t⟩⟩
- in the park ↦ λF λe [F(e) ∧ location(e, park)] ⟨⟨e,t⟩,⟨e,t⟩⟩

(1) *The murder in the park…*

↦ $\lambda F \lambda e[F(e) \wedge \text{location}(e, \text{park})]\ (\lambda e_1\ [\text{murder}(e_1)])$

(2) The fountain in the park ….

↦ $\lambda F \lambda x[F(x) \wedge \text{location}(x, \text{park})]\ (\lambda y\ [\text{fountain}(y)])$

Advantages of Davidsonian events

- ☑ Intuitive representation and semantic construction for adjuncts
- ☑ Uniform treatment of verb complements
- ☑ Uniform treatment of adjuncts and post-nominal modifiers
- ☐ Coherent treatment of tense information
- ☐ Highly compatible with analysis of semantic roles

Classical Tense Logic

- *John walks* walk(john)
- *John walked* P(walk(john))
- *John will walk* F(walk(john))

Syntax like in first-order logic, plus

- if Φ is a well-formed formula, then PΦ, FΦ, HΦ, GΦ are also well-formed formulae.

Φ has always been the case

Φ is always going to be the case

Φ happened in the past

Φ will happen in the future

Classical Tense Logic (cont.)

Tense model structures are quadruples $M = \langle U, T, <, V\rangle$ where

- U is a non-empty set of individuals (the “universe”)
- T is a non-empty sets of points in time
- $U \cap T = \varnothing$
- $<$ is a linear order on T
- V is a value assignment function, which assigns to every non-logical constant α a function from T to appropriate denotations of α

$$[\![P\Phi]\!]^{M,t,g} = 1 \text{ iff there is a } t' < t \text{ such that } [\![\Phi]\!]^{M,t',g} = 1$$

$$[\![F\Phi]\!]^{M,t,g} = 1 \text{ iff there is a } t' > t \text{ such that } [\![\Phi]\!]^{M,t',g} = 1$$

Temporal Relations and Events

(1) *The door opened, and Mary entered the room.*

(2) *John arrived. Then Mary left.*

(3) *Mary left, before John arrived.*

(4) *John arrived. Mary had left already.*

Q: How to formalize temporal relations *between events*?

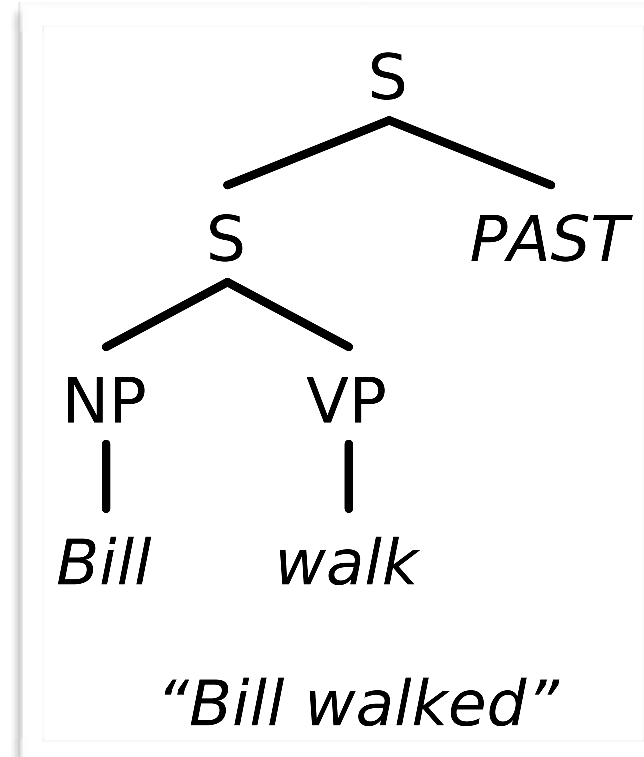
Temporal Event Structure

A model structure with events and temporal precedence is defined as $M = \langle U, E, <, e_u, V \rangle$, where

- $U \cap E = \varnothing$,
- $< \subseteq E \times E$ is an asymmetric relation (temporal precedence)
- $e_u \in E$ is the utterance event
- V is an interpretation function like in standard FOL
- Overlapping events: $e \circ e'$ iff neither $e < e'$ nor $e' < e$

Tense in Semantic Construction

We can represent inflection as an abstract tense operator reflecting the temporal location of the reported event relative to the utterance event.



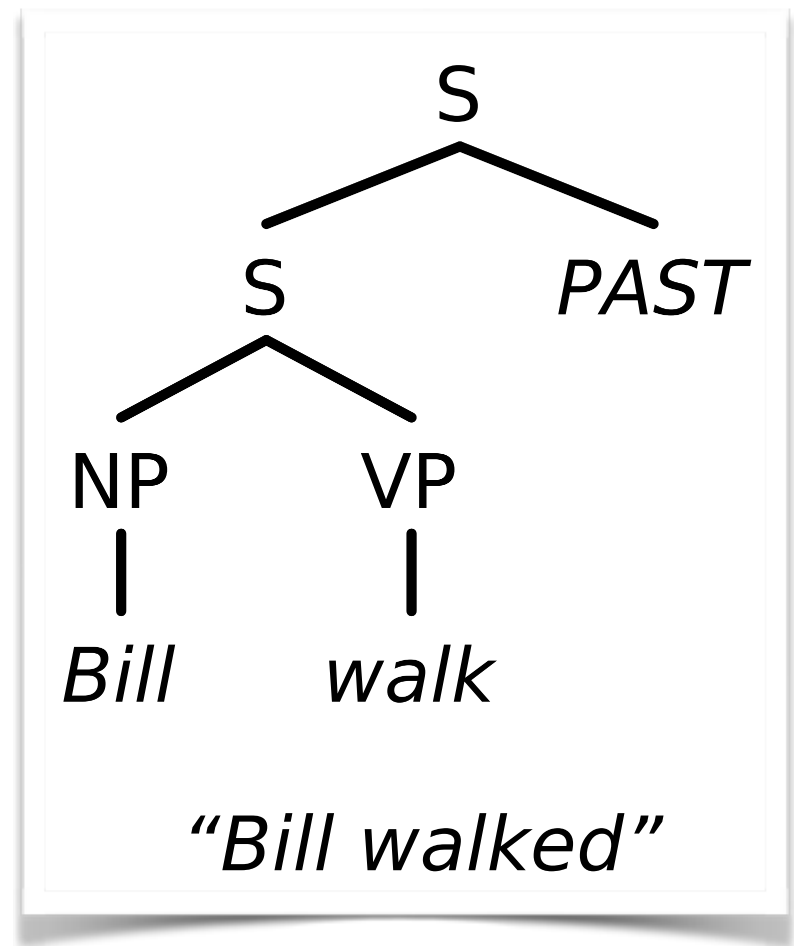
PAST $\mapsto \lambda P.\exists e\ [P(e) \wedge e < e_u] : \langle\langle e, t\rangle, t\rangle$

PRES $\mapsto \lambda P.\exists e\ [P(e) \wedge e \cdot e_u] : \langle\langle e, t\rangle, t\rangle$

Tense in Semantic Construction

Standard function application results in integration of temporal information and binding of the event variable (i.e., replacing E-CLOS):

- walk $\mapsto$ λx λe [walk(e, x)]
- Bill walk $\mapsto$ λx λe [walk(e, x)](b') $\Rightarrow^\beta$ λe [walk(e, b')]
- Bill walk PAST
  $\mapsto$ λE ∃e [E(e) ∧ e < e_u](λe' [walk(e', b)])
  $\Rightarrow^\beta$ ∃e [λe' [walk(e', b)](e) ∧ e < e_u]
  $\Rightarrow^\beta$ ∃e [walk(e, b) ∧ e < e_u]



Advantages of Davidsonian events

- ☑ Intuitive representation and semantic construction for adjuncts
- ☑ Uniform treatment of verb complements
- ☑ Uniform treatment of adjuncts and post-nominal modifiers
- ☑ Coherent treatment of tense information
- ☐ Highly compatible with analysis of semantic roles

Verbal arguments; a related problem?

(1) *John **broke** the window with a rock.*

(2) *A rock **broke** the window.*

(3) *The window **broke**.*

And we’re back to the same entailment issue:

$\exists e(\text{break}_3(e, j, w, r)) \not\models \exists e(\text{break}_2(e, r, w)) \not\models \exists e(\text{break}_1(e, w))$

Semantic/Thematic roles

agent   patient   instrument

(1) *John* ***broke*** *the window with a rock.*

$\mapsto \exists e\ [break(e) \wedge agent(e, j) \wedge patient(e, w) \wedge instrument(e, r)]$

(2) *A rock* ***broke*** *the window.*

$\mapsto \exists e\ [break(e) \wedge patient(e, w) \wedge instrument(e, r)]$

(3) *The window* ***broke****.*

$\mapsto \exists e\ [break(e) \wedge patient(e, w)]$

In standard FOL: Thematic roles are implicitly represented by the canonical order of the arguments

In Davidsonian event semantics: Thematic roles are two-place relations between the event denoted by the verb, and an argument role filler.

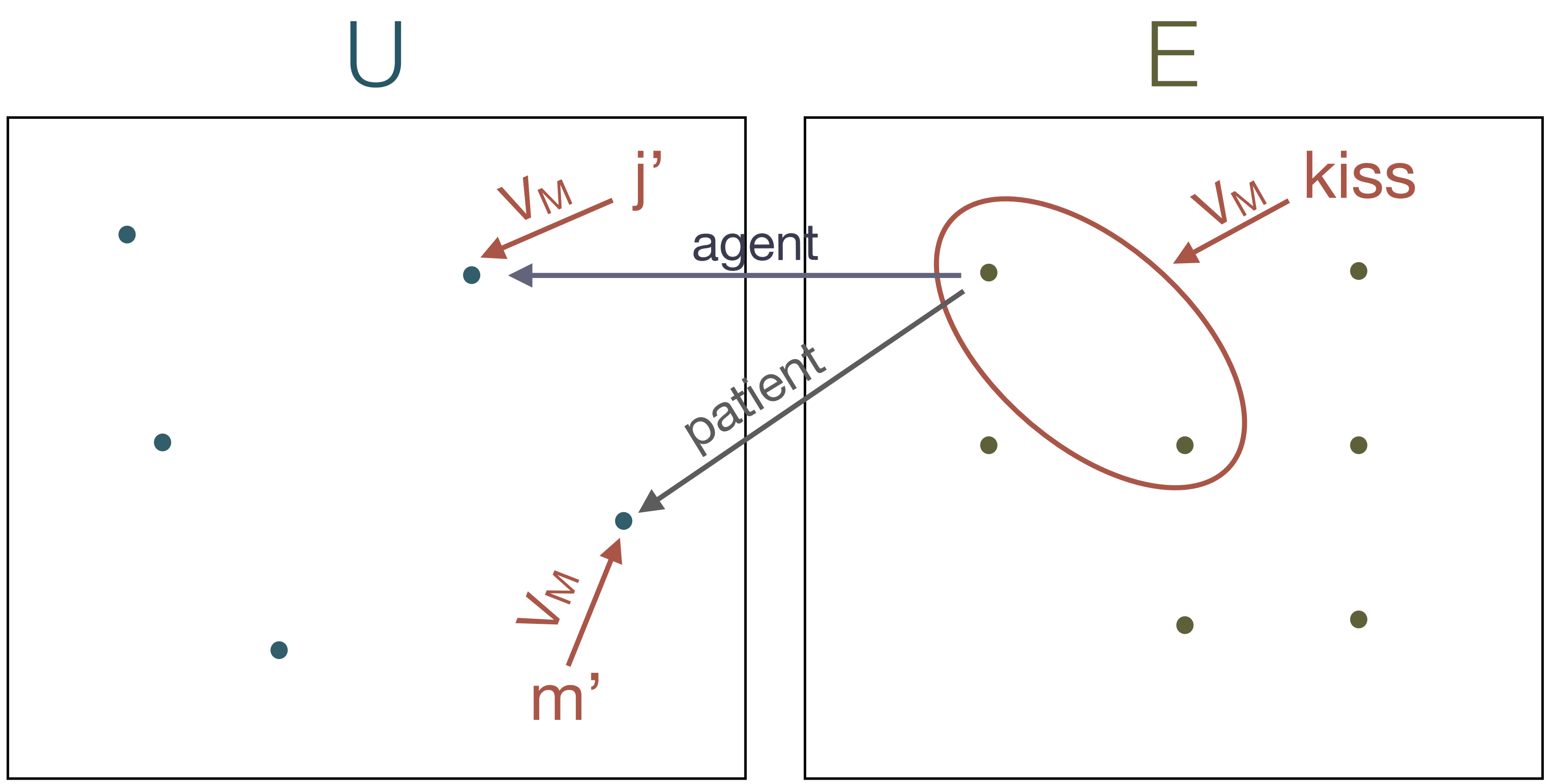
Interpretation of events with thematic roles

John kisses Mary $\mapsto$ $\exists e\,(\text{kiss}(e) \wedge \text{agent}(e, j') \wedge \text{patient}(e,m'))$

$[\![\ \exists e\,(\text{kiss}(e) \wedge \text{agent}(e, j') \wedge \text{patient}(e,m'))\ ]\!]^{M,g} = 1$

iff there is an $s \in E$ such that $[\![\text{kiss}(e)]\!]^{M,g[e/s]} = 1$ and $[\![\ \text{agent}(e, j')]\!]^{M,g[e/s]} = 1$
and $[\![\text{patient}(e,m')]\!]^{M,g[e/s]} = 1$

iff there is an $s \in E$ such that $s \in V_M(\text{kiss})$ and $\langle s, V_M(j')\rangle \in V_M(\text{agent})$
and $\langle s, V_M(m')\rangle \in V_M(\text{patient})$



Thematic roles & verbal differences/similarities

Different verbs allow different thematic role configurations

(1) *a.* *John **broke** the window with a rock* → agent, patient, instrument
b. *John **smiled** at Mary* → agent, recipient

(2) a. *The window **broke*** → allows inanimate subject
b. **The **bread** cut* → does not allow inanimate subject

Thematic roles capture equivalences and entailment relations between different predicates

(3) a. *Mary **gave** Peter the book*
b. *Peter **received** the book from Mary*

$\forall e[give(e) \leftrightarrow receive(e)] \vDash (3a) \leftrightarrow (3b)$

Determining the role inventory

Fillmore (1968): "thematic roles form a small, closed, and universally applicable inventory conceptual argument types."

A typical role inventory might consist of the roles:

- Agent, Patient, Theme, Recipient, Instrument, Source, Goal, Beneficiary, Experiencer.

But… there are some difficult cases:

(1) *Lufthansa is replacing its 737s with Airbus 320*

(2) *John sold the car to Bill for 3,000€*

(3) *Bill bought the car from John for 3,000€*

Semantic corpora with thematic roles

- Propbank: includes a separate role inventory for every lemma
- FrameNet: “Frame-based” role inventories

Frames are structured schemata representing complex prototypical situations, events, and actions

(1) *[$_{\text{Agent}}$ Lufthansa] is replacing$_{\text{Frame: REPLACING}}$ [$_{\text{Old}}$ its 737s] [$_{\text{New}}$ with Airbus A320s]*

(2) *[$_{\text{Agent}}$ Lufthansa] is substituting$_{\text{Frame: REPLACING}}$ [$_{\text{New}}$ Airbus A320s] [$_{\text{Old}}$ for its 737s]*

Semantic corpora with thematic roles (cont.)

Propbank (Palmer et al. 2005): Annotation of Penn TreeBank with predicate-argument structure.

(1) [$_{Arg0}$ Lufthansa] is replacing [$_{Arg1}$ its 737s] [$_{Arg2}$ with Airbus A320s]

Pred	replace
Arg0	Lufthansa
Arg1	its737s
Arg2	AirbusA320s

(2) [$_{Arg0}$ Lufthansa] is substituting [$_{Arg1}$ Airbus A320s] [$_{Arg2}$ for its 737s]

Pred	substitute
Arg0	Lufthansa
Arg1	AirbusA320s
Arg2	its737s

FrameNet (Baker et al. 1998): A database of frames and a lexicon with frame information

(3) [$_{Agent}$ Lufthansa] is replacing$_{\text{Frame: REPLACING}}$ [$_{Old}$ its 737s] [$_{New}$ with Airbus A320s]

(4) [$_{Agent}$ Lufthansa] is substituting$_{\text{Frame: REPLACING}}$ [$_{New}$ Airbus A320s] [$_{Old}$ for its 737s]

Frame	REPLACING
Agent	Lufthansa
Old	its737s
New	AirbusA320s

Advantages of Davidsonian events

- ☑ Intuitive representation and semantic construction for adjuncts
- ☑ Uniform treatment of verb complements
- ☑ Uniform treatment of adjuncts and post-nominal modifiers
- ☑ Plausible treatment of tense information
- ☑ Compatible with analysis of semantic roles

… but how does it combine with other semantic constructs?

A problem with events and quantification

John kissed Mary

$\mapsto \lambda P.P(j') \ [\ \lambda P.P(m')(\lambda y \lambda x \lambda e\ [kiss(e) \land agent(e,x) \land patient(e,y)])\ ]$

$\Rightarrow^{\beta} \lambda e\ [kiss(e) \land agent(e,j') \land patient(e,m')]$

$\Rightarrow^{E\text{-}CLOS} \exists e\ [kiss(e) \land agent(e,j') \land patient(e,m')]$

John kissed every girl

$\mapsto \lambda P.P(j') \ [\ \lambda P.\forall x(girl'(x) \rightarrow P(x))(\lambda y \lambda x \lambda e\ [kiss(e) \land agent(e,x) \land patient(e,y)])\ ]$

$\Rightarrow^{\beta} \lambda e\ [\forall x(girl'(x) \rightarrow kiss(e) \land agent(e,j') \land patient(e,x)]$

$\Rightarrow^{E\text{-}CLOS} \exists e\ [\forall x(girl'(x) \rightarrow kiss(e) \land agent(e,j') \land patient(e,x)]$

Two solutions to the event quantification problem

Solution I

Interpret sentences as generalized quantifiers over events: $\langle\langle e,t\rangle,t\rangle$ instead of $\langle e,t\rangle$ (E-CLOS part of lexical semantics) (Champollion, 2010; 2015)

kiss $\mapsto \lambda F_{\langle v,t\rangle}.\exists e\,(\text{kiss}(e) \wedge F(e))$ :: $\langle\langle v,t\rangle,t\rangle$ $\approx \{ F \mid F \cap \text{KISS} \neq \varnothing\}$

separate type for events!

Solution II

Introduce separate types for regular NPs and quantified NPs, and restrict existential closure to regular NPs (Winter & Zwarts, 2011; de Groote & Winter, 2014)

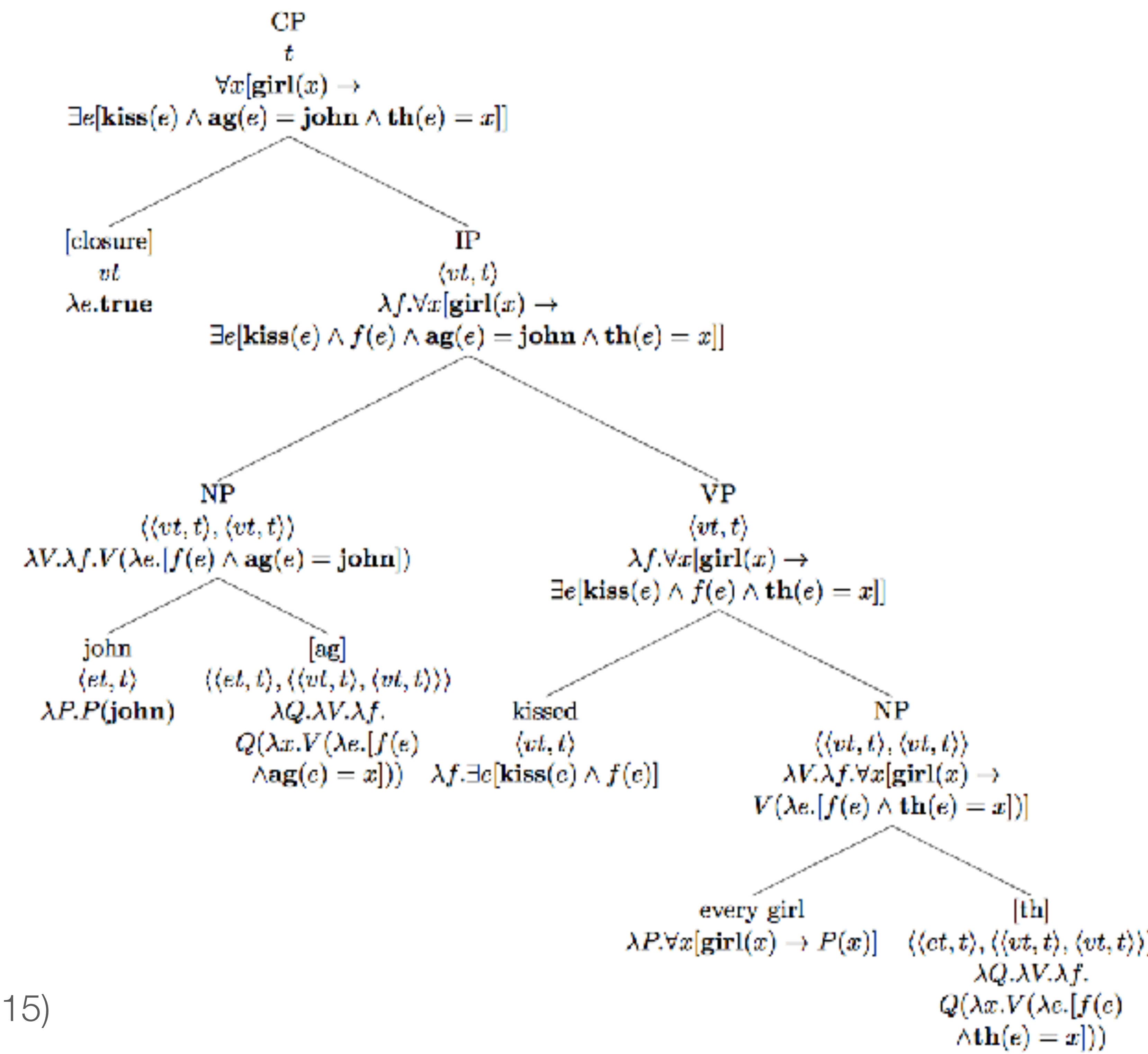
john $\mapsto$ j :: e

every girl $\mapsto \lambda P\lambda Q.\forall x(\text{girl}(x) \rightarrow Q(x))$ :: $\langle\langle e,t\rangle,\langle\langle e,t\rangle,t\rangle\rangle$

kiss $\mapsto \lambda x\lambda y\lambda e.\text{kiss}(e,x,y)$ :: $\langle e,\langle e,\langle v,t\rangle\rangle\rangle$

e-clos $\mapsto \lambda P.\exists e(P(e))$ :: $\langle\langle v,t\rangle,t\rangle$ separate type for events!

Solution I: Sentences as GQs over events



(Champollion, 2010; 2015)

Solution II: Type-restriction for existential closure

$$\frac{\vdash \textsc{every} : N \to (NP \to S) \to S \qquad \vdash \textsc{girl} : N}{\vdash \textsc{every girl} : (NP \to S) \to S}\ (1)$$

$$\frac{\dfrac{\vdash \textsc{kissed} : NP \to NP \to V \qquad x : NP \vdash x : NP}{x : NP \vdash \textsc{kissed}\, x : NP \to V} \qquad \vdash \textsc{john} : NP}{x : NP \vdash \textsc{kissed}\, x\, \textsc{john} : V}\ (2)$$

$$\frac{\dfrac{\vdash \textsc{e-clos} : V \to S \qquad \begin{array}{c}\vdots\ (2) \\ x : NP \vdash \textsc{kissed}\, x\, \textsc{john} : V\end{array}}{x : NP \vdash \textsc{e-clos}\,(\textsc{kissed}\, x\, \textsc{john}) : S}}{\vdash \lambda x.\, \textsc{e-clos}\,(\textsc{kissed}\, x\, \textsc{john}) : NP \to S}\ (3)$$

$$\frac{\begin{array}{c}\vdots\ (1) \\ \vdash \textsc{every girl} : (NP \to S) \to S\end{array} \qquad \begin{array}{c}\vdots\ (3) \\ \vdash \lambda x.\, \textsc{e-clos}\,(\textsc{kissed}\, x\, \textsc{john}) : NP \to S\end{array}}{\vdash \textsc{every girl}\,(\lambda x.\, \textsc{e-clos}\,(\textsc{kissed}\, x\, \textsc{john})) : S}$$

(Winter & Zwarts, 2011; de Groote & Winter, 2014)