Syntax-Driven Semantic Frame Composition in Lexicalized Tree Adjoining Grammars

[Tutorial]

Laura Kallmeyer & Rainer Osswald

Heinrich-Heine-Universität Düsseldorf

13th International Workshop on Tree Adjoining Grammars
and Related Formalisms (TAG+13)

September 4–6, 2017

Umeå University, Sweden

HEINRICH HEINE
UNIVERSITÄT DÜSSELDORF

CRC 991

Introduction

Goal: an LTAG architecture of the syntax-semantics interface that

- is compositional: the meaning of a complex expression can be computed from the meaning of its subparts and its composition operation.
- pairs entire elementary trees with meaning components.

Three principal approaches:

1. LTAG semantics with synchronous TAG (STAG)

   [Shieber 1994; Nesson/Shieber 2006, 2008]

2. Unification based LTAG semantics with predicate logic

   [Kallmeyer/Joshi 2003; Gardent/Kallmeyer 2003; Kallmeyer/Romero 2008]

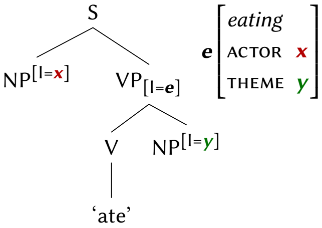
3. Unification based LTAG semantics with frames

   [Kallmeyer/Osswald 2013; Kallmeyer/Osswald/Pogodalla 2016]

This tutorial introduces the third approach.

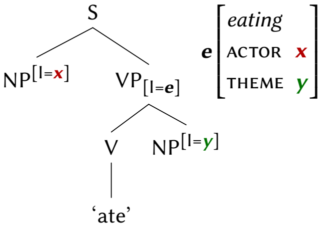
A simple example

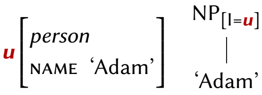
(1) Adam ate an apple.



A simple example

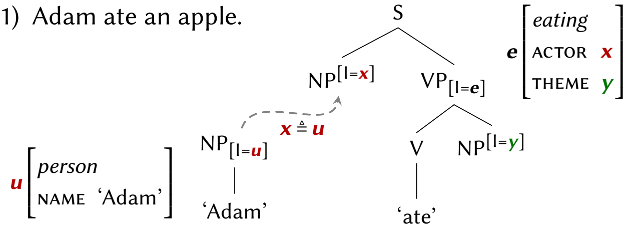
(1) Adam ate an apple.





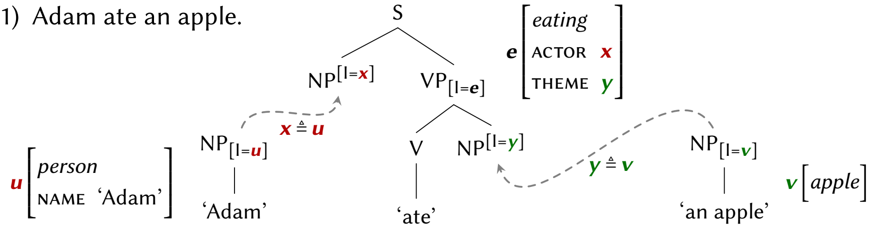
A simple example

(1) Adam ate an apple.



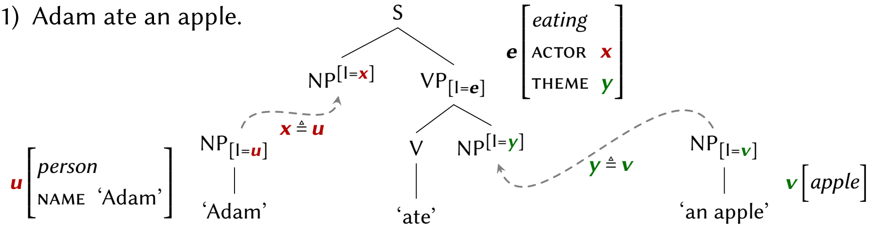
A simple example

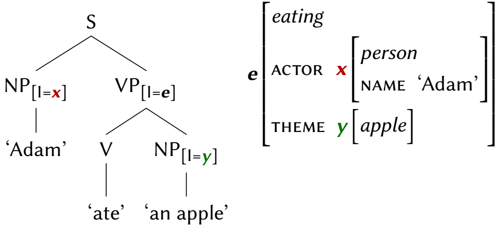
(1) Adam ate an apple.



A simple example

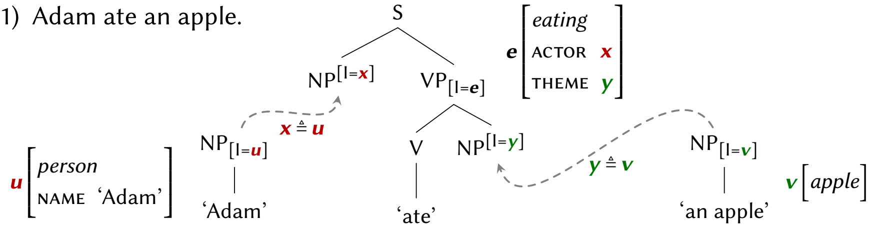
(1) Adam ate an apple.

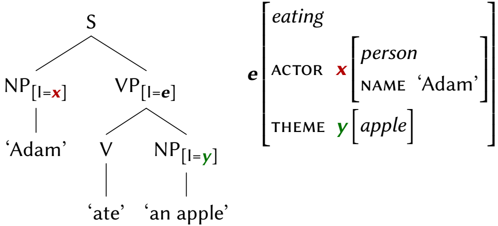


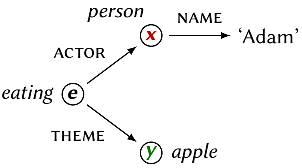


A simple example

(1) Adam ate an apple.







General properties of the syntax-semantics interface

- Semantic composition (≈ **unification**) is triggered by syntactic composition (≈ **substitution** and **adjunction**).
- Semantic representations are linked to entire elementary trees ($\rightsquigarrow$ **elementary constructions**).
- **Interface features** relate nodes in the syntactic tree to components in the semantic representation.
- Elementary constructions can be decomposed/defined by means of "metagrammatical" **constraints**.

Introduction

General properties of the syntax-semantics interface

- Semantic composition (≈ **unification**) is triggered by syntactic composition (≈ **substitution** and **adjunction**).
- Semantic representations are linked to entire elementary trees ($\rightsquigarrow$ **elementary constructions**).
- **Interface features** relate nodes in the syntactic tree to components in the semantic representation.
- Elementary constructions can be decomposed/defined by means of "metagrammatical" **constraints**.

Main components of the approach

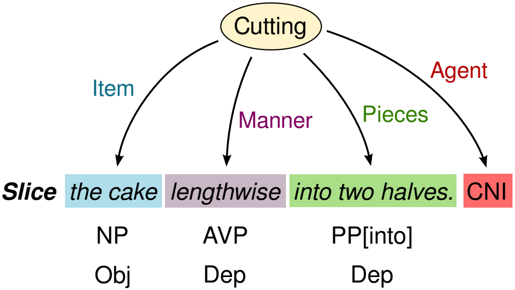
1. Lexicalized Tree Adjoining Grammars (LTAG)
   [Joshi/Schabes 1997; Abeille/Rambow 2000; ...]
2. Decompositional Frame Semantics
   [Kallmeyer/Osswald 2013; Osswald/Van Valin 2014]
3. Metagrammatical specification and decomposition
   [Crabbé/Duchier 2005; Crabbé et al. 2013, Lichte/Petitjean 2015]

Overview of the tutorial

- Frame semantics
  - Relation to semantic decomposition approaches
  - Formalization of frames and frames descriptions
  - Subsumption and unification
- Recap of Lexicalized Tree Adjoining Grammars (LTAG)
  - Basic definitions; feature-structure based TAG
  - Key properties of LTAG: extended domain of locality, etc.
  - Tree families and lexical anchoring
- Specification of elementary trees in the metagrammar
- Combing LTAG and frame semantics
  - Elementary constructions
  - Specification of constructions in the metagrammar
  - Examples (caused motion, Dative alternation)
- XMG implementation of frame semantics
- Frame semantics and quantification
- Further topics

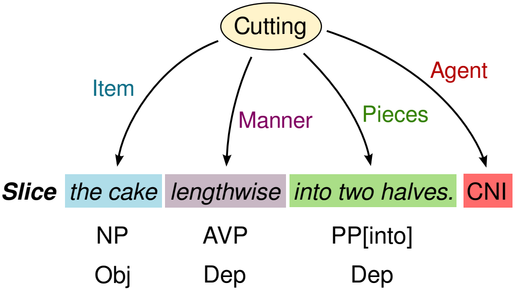
Frame semantics: introduction

- Frames à la Fillmore/FrameNet



Frame semantics: introduction

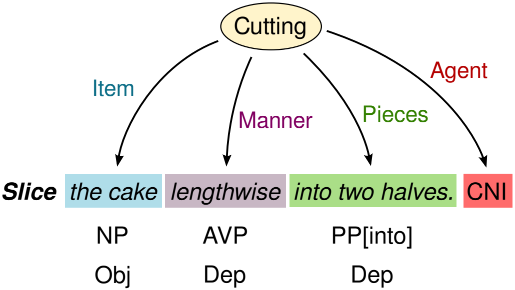
- Frames à la Fillmore/FrameNet



- Frames (possibly embedded attribute-value structures with constraints) as a general format of conceptual representation

[Barsalou 1992; Löbner 2014]

- Frames à la Fillmore/FrameNet



- Frames (possibly embedded attribute-value structures with constraints) as a general format of conceptual representation

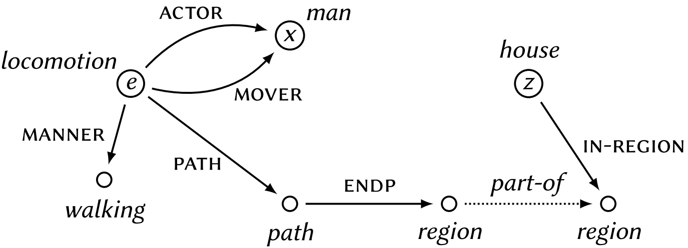
[Barsalou 1992; Löbner 2014]

- Decompositional frame semantics

[Kallmeyer/Osswald 2013; Osswald/Van Valin 2014]

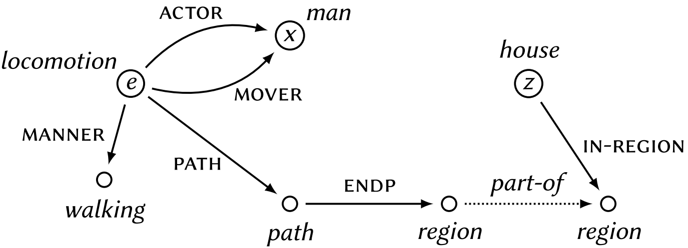
Frame semantics: introduction

- Semantic frames are commonly depicted as **graphs** with labeled nodes and edges, where **nodes** correspond to entities (individuals, events, …) and **edges** to functional (or non-functional) relations between these entities.



Frame semantics: introduction

- Semantic frames are commonly depicted as **graphs** with labeled nodes and edges, where **nodes** correspond to entities (individuals, events, …) and **edges** to functional (or non-functional) relations between these entities.



- Frames in this sense can be formalized as generalized feature structures with types, relations and node labels.

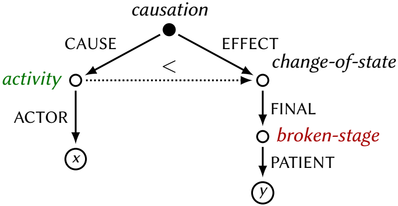
Example Lexical decomposition templates [Rappaport Hovav/Levin 1998]

(2) [[*x* ACT] CAUSE [BECOME [*y* *BROKEN*]]]

Frame semantics: introduction

Example Lexical decomposition templates [Rappaport Hovav/Levin 1998]

(2) [[*x* ACT] CAUSE [BECOME [*y* *BROKEN*]]]



$$\begin{bmatrix} \textit{causation} & \\ \text{CAUSE} & \begin{bmatrix} \textit{activity} & \\ \text{EFFECTOR} & x \end{bmatrix} \\ \text{EFFECT} & \begin{bmatrix} \textit{change-of-state} & \\ \text{FINAL} & \begin{bmatrix} \textit{broke-stage} & \\ \text{PATIENT} & y \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\text{CAUSE} < \text{EFFECT}$$

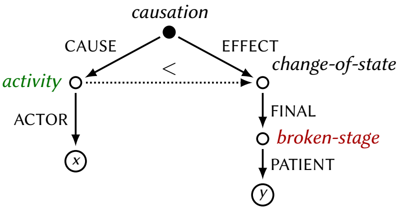
Description in attribute-value logic

$$\textit{causation} \wedge \text{CAUSE} : \textit{activity} \wedge \text{CAUSE ACTOR} \triangleq x$$
$$\wedge\ \text{EFFECT}\,(\textit{change-of-state} \wedge \text{FINAL} : (\textit{broken-stage} \wedge \text{PATIENT} \triangleq y))$$
$$\wedge\ \text{CAUSE} < \text{EFFECT}$$

Frame semantics: introduction

Example Lexical decomposition templates [Rappaport Hovav/Levin 1998]

(2) [[*x* ACT] CAUSE [BECOME [*y* *BROKEN*]]]



$$\begin{bmatrix} \textit{causation} & \\ \text{CAUSE} & \begin{bmatrix}\textit{activity} & \\ \text{EFFECTOR} & x\end{bmatrix} \\ \text{EFFECT} & \begin{bmatrix}\textit{change-of-state} & \\ \text{FINAL} & \begin{bmatrix}\textit{broke-stage} & \\ \text{PATIENT} & y\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

$\text{CAUSE} < \text{EFFECT}$

Description in attribute-value logic

$$\textit{causation} \wedge \text{CAUSE} : \textit{activity} \wedge \text{CAUSE ACTOR} \triangleq x$$
$$\wedge\ \text{EFFECT}\,(\textit{change-of-state} \wedge \text{FINAL} : (\textit{broken-stage} \wedge \text{PATIENT} \triangleq y))$$
$$\wedge\ \text{CAUSE} < \text{EFFECT}$$

Translation into first-order logic

$$\lambda e \exists e' \exists e'' \exists s(\textit{causation}(e) \wedge \text{CAUSE}(e, e') \wedge \text{EFFECT}(e, e'') \wedge e' < e'' \wedge$$
$$\textit{activity}(e') \wedge \text{ACTOR}(e', x) \wedge \textit{change-of-state}(e'') \wedge$$
$$\text{FINAL}(e'', s) \wedge \textit{broken-stage}(s) \wedge \text{PATIENT}(s, y))$$

Frame semantics: introduction

Basic assumptions

- **Attributes** (features, functional roles/relations) play a central role in the organization of semantic and conceptual knowledge and representation. [Barsalou 1992; Löbner 2014]

- Semantic components (participants, subevents) can be (recursively) addressed by **attributes**.

  ⇝ inherently **structured representations** (models); composition by **unification** (under constraints)

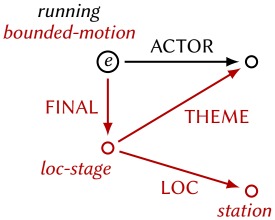
- Semantic processing may be seen as the **incremental construction** of **minimal** (**frame**) **models** based on the input, the context, and background knowledge (lexicon, …).

Frame semantics: introduction

Example

(3) Anna ran to the station.

$$e \begin{bmatrix} \textit{running} \wedge \textit{bounded-motion} & \\ \text{ACTOR} & \boxed{1} \\ \text{FINAL} & \begin{bmatrix} \textit{loc-stage} & \\ \text{THEME} & \boxed{1} \\ \text{LOC} & \textit{station} \end{bmatrix} \end{bmatrix}$$



Attribute-value logic

$$e \cdot (\textit{running} \wedge \textit{bounded-motion} \wedge \text{ACTOR} \doteq \text{FINAL THEME}$$
$$\wedge\ \text{FINAL} : (\textit{loc-stage} \wedge \text{LOC} : \textit{station}))$$

Translation into first-order logic

$$\exists x \exists s \exists y (\textit{running}(e) \wedge \textit{bounded-motion}(e) \wedge \text{ACTOR}(e, x)$$
$$\wedge\ \text{FINAL}(e, s) \wedge \textit{loc-stage}(s) \wedge \text{THEME}(s, x) \wedge \text{LOC}(s, y) \wedge \textit{station}(y))$$

Constraints

$\textit{running} \Rightarrow \textit{activity}$ (short for $\forall e(\textit{running}(e) \rightarrow \textit{activity}(e))$),

$\textit{loc-stage} \Rightarrow \text{THEME} : \top \wedge \text{LOC} : \top, \quad \ldots$

Frame semantics: formalization

Vocabulary / Signature

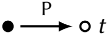
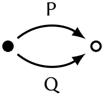
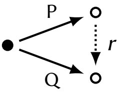
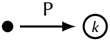
Attr	attributes (= dyadic functional relation symbols)
Rel	(proper) relation symbols
Type	type symbols (= monadic predicates)
Nname	node names ("nominals") } Nlabel
Nvar	node variables } Nlabel

Primitive attribute-value descriptions (pAVDesc)

$$t \mid p : t \mid p \doteq q \mid [p_1, \ldots, p_n] : r \mid p \triangleq k$$

($t \in$ Type, $r \in$ Rel, $p, q, p_i \in$ Attr*, $k \in$ Nlabel)

Semantics

Description	Graph	AVM
$\mathrm{P} : t$		$\begin{bmatrix} \mathrm{P} & t \end{bmatrix}$
$\mathrm{P} \doteq \mathrm{Q}$		$\begin{bmatrix} \mathrm{P} & \boxed{1} \\ \mathrm{Q} & \boxed{1} \end{bmatrix}$
$[\mathrm{P}, \mathrm{Q}] : r$		$\begin{bmatrix} \mathrm{P} & \boxed{1} \\ \mathrm{Q} & \boxed{2} \end{bmatrix}$ $r(\boxed{1}, \boxed{2})$
$\mathrm{P} \triangleq k$		$\begin{bmatrix} \mathrm{P} & k[\ ] \end{bmatrix}$

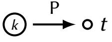
Primitive attribute-value formulas (pAVForm)

$$k \cdot p : t \;\mid\; k \cdot p \triangleq l \cdot q \;\mid\; \langle k_1 \cdot p_1, \ldots, k_n \cdot p_n \rangle : r$$

($t \in \text{Type}$, $r \in \text{Rel}$, $p, q, p_i \in \text{Attr}^*$, $k, l, k_i \in \text{Nlabel}$)

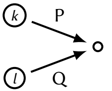
Semantics

$k \cdot \mathrm{P} : t$



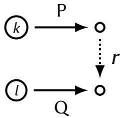
$k \begin{bmatrix} \mathrm{P} & t \end{bmatrix}$

$k \cdot \mathrm{P} \triangleq l \cdot \mathrm{Q}$



$k \begin{bmatrix} \mathrm{P} & \boxed{1} \end{bmatrix}$
$l \begin{bmatrix} \mathrm{Q} & \boxed{1} \end{bmatrix}$

$\langle k \cdot \mathrm{P}, l \cdot \mathrm{Q} \rangle : r$



$k \begin{bmatrix} \mathrm{P} & \boxed{1} \end{bmatrix}$
$l \begin{bmatrix} \mathrm{Q} & \boxed{2} \end{bmatrix}$
$r(\boxed{1}, \boxed{2})$

Frame semantics: formalization

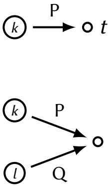
Primitive attribute-value formulas (pAVForm)

$$k \cdot p\!: t \;\mid\; k \cdot p \triangleq l \cdot q \;\mid\; \langle k_1 \cdot p_1, \ldots, k_n \cdot p_n \rangle : r$$

($t \in$ Type, $r \in$ Rel, $p, q, p_i \in \text{Attr}^*$, $k, l, k_i \in$ Nlabel)

Semantics

$k \cdot \mathrm{P} : t$

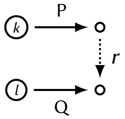


$k\begin{bmatrix}\mathrm{P} & t\end{bmatrix}$

$k \cdot \mathrm{P} \triangleq l \cdot \mathrm{Q}$

$k\begin{bmatrix}\mathrm{P} & \boxed{1}\end{bmatrix}$ $l\begin{bmatrix}\mathrm{Q} & \boxed{1}\end{bmatrix}$

$\langle k \cdot \mathrm{P}, l \cdot \mathrm{Q} \rangle : r$



$k\begin{bmatrix}\mathrm{P} & \boxed{1}\end{bmatrix}$ $l\begin{bmatrix}\mathrm{Q} & \boxed{2}\end{bmatrix}$ $r(\boxed{1}, \boxed{2})$

Formal definitions (fairly standard)

Set/universe of "nodes"	V
Interpretation function	$\mathcal{I}$: Attr $\to [V \rightharpoonup V]$, Type $\to \wp(V)$,
	Rel $\to \bigcup_n \wp(V^n)$, Nname $\to V$
(Partial) variable assignment	g : Nvar $\rightharpoonup V$

Frame semantics: formalization

Formal definitions (cont'd)

Abbreviation: $\mathcal{I}_g(k) = v$ for $k \in$ Nlabel iff $\mathcal{I}(k) = v$ if $k \in$ Nname and
$g(k) = v$ if $k \in$ Nvar ($g(k)$ defined)

Satisfaction of primitive descriptions

$$
\begin{array}{ll}
\langle V, \mathcal{I}, g \rangle, v \vDash t & \text{iff } v \in \mathcal{I}(t) \\
\langle V, \mathcal{I}, g \rangle, v \vDash p : t & \text{iff } \mathcal{I}(p)(v) \in \mathcal{I}(t) \\
\langle V, \mathcal{I}, g \rangle, v \vDash p \doteq q & \text{iff } \mathcal{I}(p)(v) = \mathcal{I}(q)(v) \\
\langle V, \mathcal{I}, g \rangle, v \vDash [p_1, \ldots, p_n] : r & \text{iff } \langle \mathcal{I}(p_1)(v), \ldots, \mathcal{I}(p_n)(v) \rangle \in \mathcal{I}(r) \\
\langle V, \mathcal{I}, g \rangle, v \vDash p \triangleq k & \text{iff } \mathcal{I}(p)(v) = \mathcal{I}_g(k) \quad (k \in \text{Nlabel})
\end{array}
$$

Satisfaction of primitive formulas

$$
\begin{array}{ll}
\langle V, \mathcal{I}, g \rangle \vDash k \cdot p : t & \text{iff } \mathcal{I}(p)(\mathcal{I}_g(k)) \in \mathcal{I}(t) \\
\langle V, \mathcal{I}, g \rangle \vDash k \cdot p \triangleq l \cdot q & \text{iff } \mathcal{I}(p)(\mathcal{I}_g(k)) = \mathcal{I}(q)(\mathcal{I}_g(l)) \\
\langle V, \mathcal{I}, g \rangle \vDash \langle k_1 \cdot p_1, \ldots, k_n \cdot p_n \rangle : r & \text{iff } \langle \mathcal{I}(p_1)(\mathcal{I}_g(k_1)), \ldots, \mathcal{I}_g(p_n)(\mathcal{I}(k_n)) \rangle \in \mathcal{I}(r)
\end{array}
$$

Satisfaction of Boolean combinations as usual.

Frame semantics: formalization

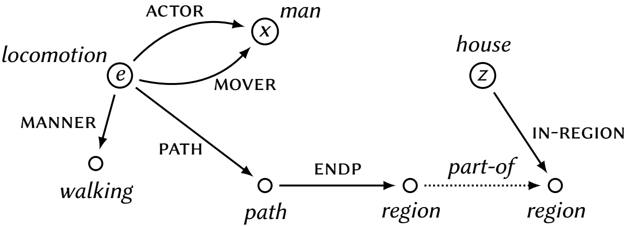
Frame F over $\langle$Attr, Type, Rel, Nname, Nvar$\rangle$:

$F = \langle V, \mathcal{I}, g \rangle$, with V finite, such that every node $v \in V$ is reachable from some labeled node $w \in V$ via an attribute path, i.e.,

(i) $w = \mathcal{I}_g(k)$ for some $k \in$ Nlabel (= Nname $\cup$ Nvar) and

(ii) $v = \mathcal{I}(p)(w)$ for some $p \in$ Attr*.

Example



A frame $F = \langle V, \mathcal{I}, g \rangle$ is a **model** of an attribute-value formula ϕ iff $F \vDash \phi$.

Subsumption

$F_1 = \langle V_1, \mathcal{I}_1, g_1 \rangle$ **subsumes** $F_2 = \langle V_2, \mathcal{I}_2, g_2 \rangle$ ($F_1 \sqsubseteq F_2$) iff there is a (necessarily unique) **morphism** $h : F_1 \to F_2$, i.e., a function $h : V_1 \to V_2$ such that

(i) $\mathcal{I}_2(f)(h(v)) = h(\mathcal{I}_1(f)(v))$, if $\mathcal{I}_1(f)(v)$ is defined, $f \in$ Attr, $v \in V_1$,

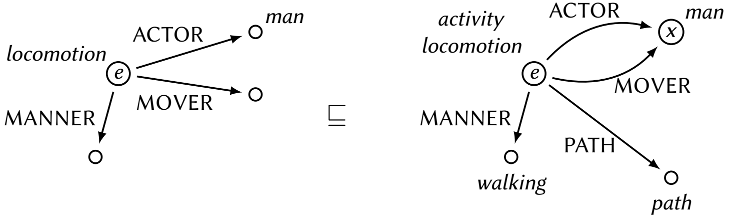
(ii) $h(\mathcal{I}_1(t)) \subseteq \mathcal{I}_2(t)$, for $t \in$ Type

(iii) $h(\mathcal{I}_1(r)) \subseteq \mathcal{I}_2(r)$, for $r \in$ Rel

(iv) $h(\mathcal{I}_1(n)) = \mathcal{I}_2(n)$, for $n \in$ Nname

(v) $h(g_1(x)) = g_2(x)$, for $x \in$ Nvar, if $g_1(x)$ is defined

Example



Frame semantics: formalization

Subsumption

$F_1 = \langle V_1, \mathcal{I}_1, g_1 \rangle$ **subsumes** $F_2 = \langle V_2, \mathcal{I}_2, g_2 \rangle$ ($F_1 \sqsubseteq F_2$) iff there is a (necessarily unique) **morphism** $h : F_1 \to F_2$, i.e., a function $h : V_1 \to V_2$ such that

(i) $\mathcal{I}_2(f)(h(v)) = h(\mathcal{I}_1(f)(v))$, if $\mathcal{I}_1(f)(v)$ is defined, $f \in$ Attr, $v \in V_1$,

(ii) $h(\mathcal{I}_1(t)) \subseteq \mathcal{I}_2(t)$, for $t \in$ Type

(iii) $h(\mathcal{I}_1(r)) \subseteq \mathcal{I}_2(r)$, for $r \in$ Rel

(iv) $h(\mathcal{I}_1(n)) = \mathcal{I}_2(n)$, for $n \in$ Nname

(v) $h(g_1(x)) = g_2(x)$, for $x \in$ Nvar, if $g_1(x)$ is defined

Unification

Least upper bound $F_1 \sqcup F_2$ of F_1 and F_2 w.r.t. subsumption.

Frame semantics: formalization

Subsumption

$F_1 = \langle V_1, \mathcal{I}_1, g_1 \rangle$ **subsumes** $F_2 = \langle V_2, \mathcal{I}_2, g_2 \rangle$ ($F_1 \sqsubseteq F_2$) iff there is a (necessarily unique) **morphism** $h : F_1 \to F_2$, i.e., a function $h : V_1 \to V_2$ such that

(i) $\mathcal{I}_2(f)(h(v)) = h(\mathcal{I}_1(f)(v))$, if $\mathcal{I}_1(f)(v)$ is defined, $f \in \text{Attr}$, $v \in V_1$,

(ii) $h(\mathcal{I}_1(t)) \subseteq \mathcal{I}_2(t)$, for $t \in \text{Type}$

(iii) $h(\mathcal{I}_1(r)) \subseteq \mathcal{I}_2(r)$, for $r \in \text{Rel}$

(iv) $h(\mathcal{I}_1(n)) = \mathcal{I}_2(n)$, for $n \in \text{Nname}$

(v) $h(g_1(x)) = g_2(x)$, for $x \in \text{Nvar}$, if $g_1(x)$ is defined

Unification

Least upper bound $F_1 \sqcup F_2$ of F_1 and F_2 w.r.t. subsumption.

Theorem (Frame unification) [≈ Hegner 1994]

The worst case time-complexity of frame unification is almost linear in the number of nodes.

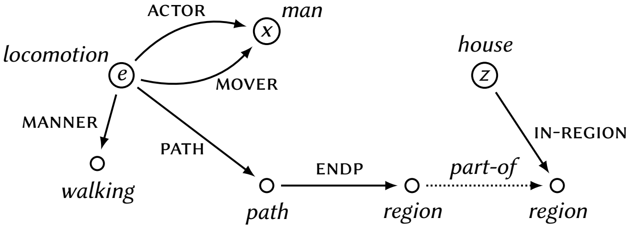
Frame semantics: formalization

Frames as minimal models of attribute-value formulas

(i) Every frame is the minimal model (w.r.t. subsumption) of a finite conjunction of primitive attribute-value formulas.

(ii) Every finite conjunction of primitive attribute-value formulas has a minimal frame model.

Example



$$e \cdot (\textit{locomotion} \wedge \textsc{manner}: \textit{walking} \wedge \textsc{actor} \triangleq x$$
$$\wedge\ \textsc{mover} \doteq \textsc{actor} \wedge \textsc{path}: (\textit{path} \wedge \textsc{endp}: \textit{region}))$$
$$\wedge\ \langle e \cdot \textsc{path endp}, z \cdot \textsc{in-region} \rangle : \textit{part-of} \ \wedge\ x \cdot \textit{man}$$

Frame semantics: formalization

Attribute-value constraints

General format: $\forall\phi,\ \phi \in$ AVDesc $(\langle V, \mathcal{I}, g\rangle \vDash \forall\phi$ iff $\langle V, \mathcal{I}, g\rangle, v \vDash \phi$ for every $v \in V)$

Notation: $\phi \Rightarrow \psi$ for $\forall(\phi \rightarrow \psi)$

Horn constraints: $\phi_1 \wedge \ldots \wedge \phi_n \Rightarrow \psi$ $(\phi_i \in \text{pAVDesc} \cup \{\top\}, \psi \in \text{pAVDesc} \cup \{\bot\})$

Examples

$\textit{activity} \Rightarrow \textit{event}$
$\textit{causation} \wedge \textit{activity} \Rightarrow \bot$
$\text{AGENT} : \top \Rightarrow \text{AGENT} \doteq \text{ACTOR}$
$\textit{activity} \Rightarrow \text{ACTOR} : \top$
$\textit{activity} \wedge \textit{motion} \Rightarrow \text{ACTOR} \doteq \text{MOVER}$
...

Frame semantics: formalization

Attribute-value constraints

General format: $\forall\phi,\ \phi \in \text{AVDesc}$ $(\langle V, \mathcal{I}, g\rangle \vDash \forall\phi$ iff $\langle V, \mathcal{I}, g\rangle, v \vDash \phi$ for every $v \in V)$

Notation: $\phi \Rightarrow \psi$ for $\forall(\phi \rightarrow \psi)$

Horn constraints: $\phi_1 \wedge \ldots \wedge \phi_n \Rightarrow \psi$ $(\phi_i \in \text{pAVDesc} \cup \{\top\}, \psi \in \text{pAVDesc} \cup \{\bot\})$

Examples

activity $\Rightarrow$ *event*

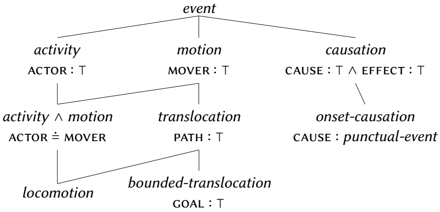
causation $\wedge$ *activity* $\Rightarrow \bot$

AGENT : $\top \Rightarrow$ AGENT $\doteq$ ACTOR

activity $\Rightarrow$ ACTOR : $\top$

activity $\wedge$ *motion* $\Rightarrow$ ACTOR $\doteq$ MOVER

…



Frame semantics: formalization

Attribute-value constraints

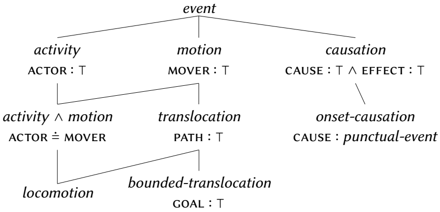
General format: $\forall\phi,\ \phi \in \text{AVDesc}$ ($\langle V, \mathcal{I}, g\rangle \vDash \forall\phi$ iff $\langle V, \mathcal{I}, g\rangle, v \vDash \phi$ for every $v \in V$)

Notation: $\phi \Rightarrow \psi$ for $\forall(\phi \rightarrow \psi)$

Horn constraints: $\phi_1 \wedge \ldots \wedge \phi_n \Rightarrow \psi$ ($\phi_i \in \text{pAVDesc} \cup \{\top\}$, $\psi \in \text{pAVDesc} \cup \{\bot\}$)

Examples

activity $\Rightarrow$ *event*
causation $\wedge$ *activity* $\Rightarrow \bot$
AGENT : $\top \Rightarrow$ AGENT $\doteq$ ACTOR
activity $\Rightarrow$ ACTOR : $\top$
activity $\wedge$ *motion* $\Rightarrow$ ACTOR $\doteq$ MOVER
…



Theorem (Frame unification under Horn constraints) [≈ Hegner 1994]

The worst case time-complexity of frame unification under a finite set of labeled Horn constraints is almost linear in the number of nodes.

(Labeled Horn constraint: $k_1 \cdot \phi_1 \wedge \ldots \wedge k_n \cdot \phi_n \rightarrow l \cdot \psi$)

Further examples [Babonnaud et al. 2016]

book ⇛ *info-carrier*

Frame semantics: formalization

Further examples [Babonnaud et al. 2016]

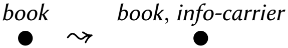
book ⇒ *info-carrier*

book ● $\rightsquigarrow$ *book*, *info-carrier* ●

Frame semantics: formalization

Further examples [Babonnaud et al. 2016]

book $\Rightarrow$ *info-carrier*

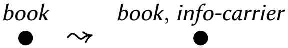


info-carrier $\Rightarrow$ *phys-obj* $\wedge$ CONTENT : *information*

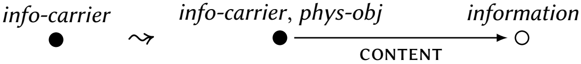
Frame semantics: formalization

Further examples [Babonnaud et al. 2016]

book ⇒ *info-carrier*

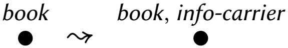


info-carrier ⇒ *phys-obj* ∧ CONTENT : *information*

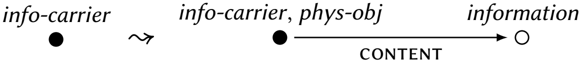


Further examples [Babonnaud et al. 2016]

book ⇒ *info-carrier*



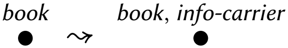
info-carrier ⇒ *phys-obj* ∧ CONTENT : *information*



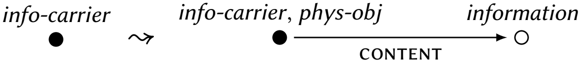
reading ⇒ PERC-COMP : *perception* ∧ MENT-COMP : *comprehension*
∧ [PERC-COMP, MENT-COMP] : *ordered-overlap*

Further examples [Babonnaud et al. 2016]

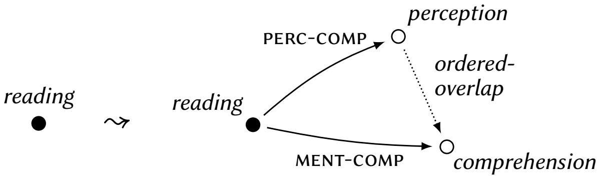
book ⇒ *info-carrier*



info-carrier ⇒ *phys-obj* ∧ CONTENT : *information*



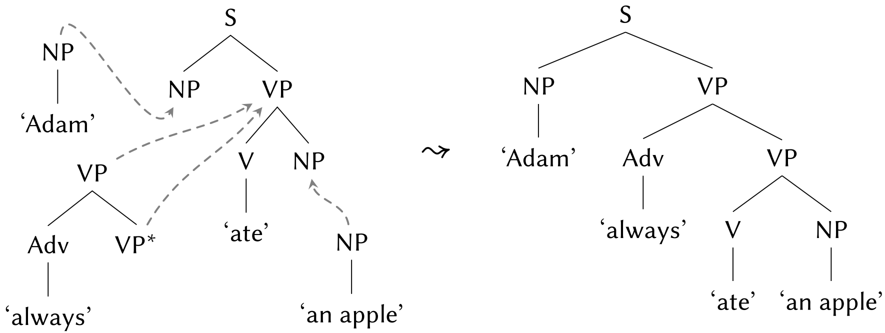
reading ⇒ PERC-COMP : *perception* ∧ MENT-COMP : *comprehension*
∧ [PERC-COMP, MENT-COMP] : *ordered-overlap*



Lexicalized Tree Adjoining Grammars (LTAG): recap

Tree-rewriting system [Joshi/Schabes 1997; Abeille/Rambow 2000; ...]

- Finite set of (**lexicalized**) **elementary trees**.
- Two operations: **substitution** (replacing a leaf with a new tree) and **adjunction** (replacing an internal node with a new tree).



Lexicalized Tree Adjoining Grammars (LTAG): recap

Feature-structure based TAG (FTAG) [Vijay-Shanker/Joshi 1988]

Each node has a top and a bottom feature structure:

- The top feature structure provide information about what the node presents within the surrounding structure.
- The bottom feature structure provide information about what the tree below the node represents.

In the final derived tree, top and bottom must be unified.

Operations on feature structures under substitution:

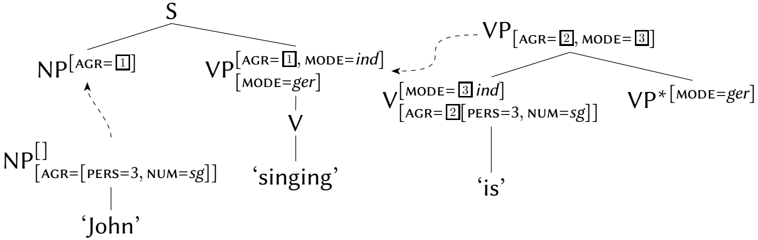
- The top of the root of the new initial tree unifies with the top of the substitution node.

Operations on feature structures under adjunction:

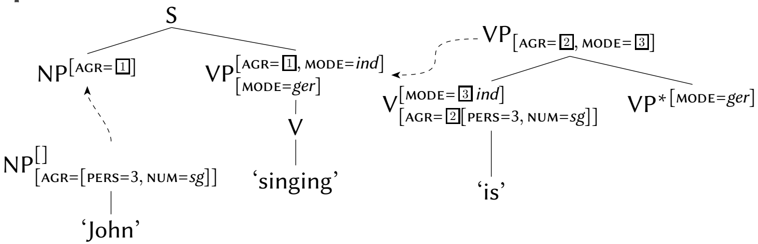
- The top of the root of the new auxiliary tree unifies with the top of the adjunction site; the bottom of the foot of the new tree unifies with the bottom of the adjunction site.

Lexicalized Tree Adjoining Grammars (LTAG): recap

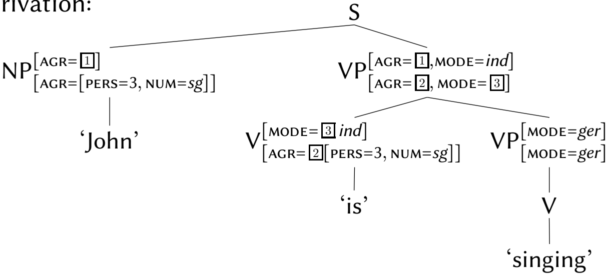
Example



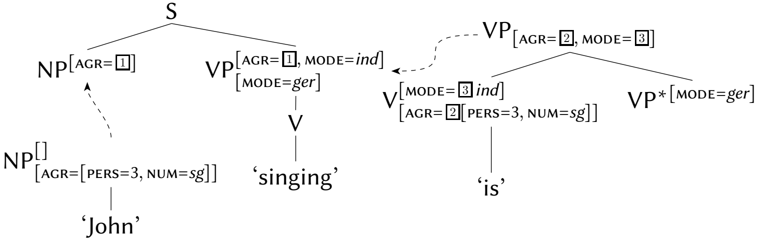
Example



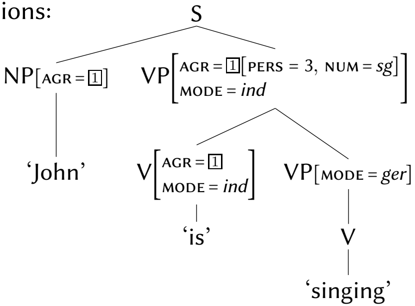
Result of derivation:



Example



After top-bottom unifications:



Lexicalized Tree Adjoining Grammars (LTAG): recap

Two key properties of the LTAG formalism

- **Extended domain of locality**

  The full argument projection of a lexical item can be represented by a single elementary tree.

  Elementary trees can have a complex constituent structure.

- **Factoring recursion from the domain of dependencies**

  Constructions related to iteration and recursion are modeled by adjunction.

  Through adjunction, the local dependencies encoded by elementary trees can become long-distance dependencies in the derived trees.

Lexicalized Tree Adjoining Grammars (LTAG): recap

Two key properties of the LTAG formalism

- **Extended domain of locality**

  The full argument projection of a lexical item can be represented by a single elementary tree.

  Elementary trees can have a complex constituent structure.

- **Factoring recursion from the domain of dependencies**

  Constructions related to iteration and recursion are modeled by adjunction.

  Through adjunction, the local dependencies encoded by elementary trees can become long-distance dependencies in the derived trees.

Slogan: "**Complicate locally**, **simplify globally**" [Bangalore/Joshi 2010]

Lexicalized Tree Adjoining Grammars (LTAG): recap

“**Simplify globally**”

- The composition of elementary trees can be expressed by two general operations: substitution and adjunction.

  (Since basically all linguistic constraints are specified over the local domains represented by elementary trees.)

“**Complicate locally**”

- Elementary trees can have complex semantic representations which are not necessarily derived compositionally (in the syntax) from smaller parts of the trees.

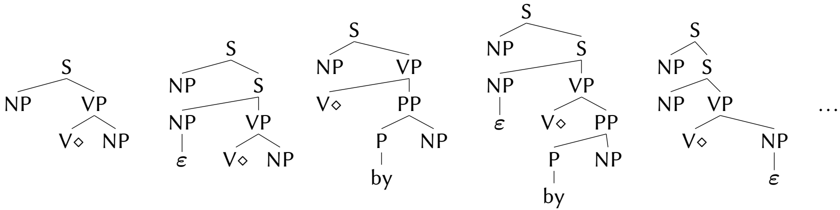
  In particular, there is no need to reproduce the internal structure of an elementary syntactic tree within its associated semantic representation. [Kallmeyer/Joshi 2003]

Lexicalized Tree Adjoining Grammars (LTAG): recap

Tree families

Unanchored elementary trees are organized in tree families, which capture variations in the (syntactic) subcategorization frames.

Example unanchored tree family for transitive verbs

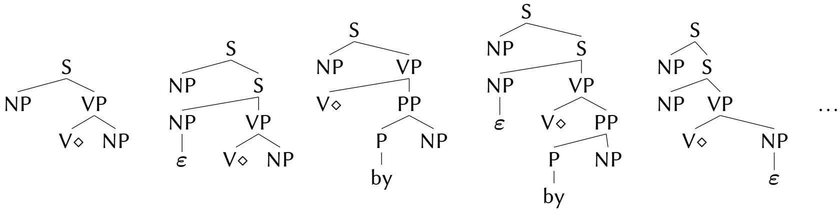


Lexicalized Tree Adjoining Grammars (LTAG): recap

Tree families

Unanchored elementary trees are organized in tree families, which capture variations in the (syntactic) subcategorization frames.

Example unanchored tree family for transitive verbs



Options for the specification/generation of tree families:

- Transformation rules applied to base trees (e.g., metarules in XTAG)
- Classes of tree constraints ("metagrammar", XMG system)

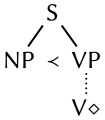
LTAG & metagrammar specification

XMG (eXtensible MetaGrammar) [e.g., Crabbé et al. 2013]

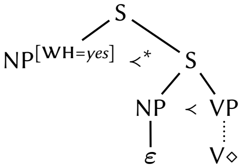
- A framework for specifying (the elementary structures of) tree based grammars by means of a **declarative** language (e.g., by dominance and precedence constraints)
- The specifications are organized into **classes** that can be **reused** ("imported") by other classes.
- Classes may contain descriptions from different **dimensions**, and the XMG system can be extended in this respect, e.g., by a dimension of **frame** descriptions.
- An XMG **compiler** generates the elementary structures of a grammar from a metagrammar.

Example

Class *CanSubj*



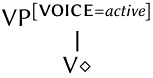
Class *ExtrSubj*



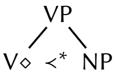
Class *Subj*

$CanSubj \vee ExtSubj$

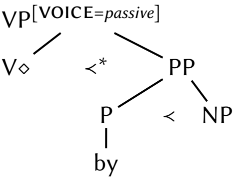
Class *ActV*



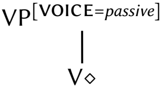
Class *DirObj*



Class *ByObj*



Class *PassV*

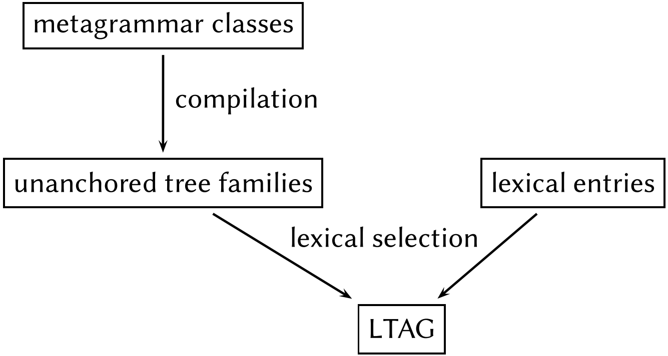


Class *Transitive*

$((Subj \wedge ActV) \vee ByObj \vee PassV) \wedge ((DirObj \wedge ActV) \vee (Subj \wedge PassV))$

LTAG & metagrammar specification

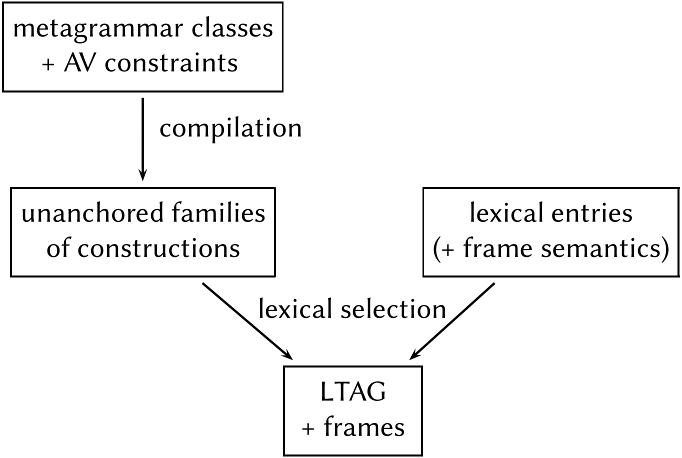
Overall architecture



Next step:

Add (frame) semantics to all components and link syntax to semantics.

Overall architecture (syntax + semantics)



Elements of the syntax-semantics interface

- **Elementary construction:**

  elementary tree
  \+ semantic frame
  \+ linking of frame node variables to index features in the tree

- Specification in the metagrammar:

  classes of tree constraints
  \+ sets of attribute-value constraints
  \+ linking of variables to index features

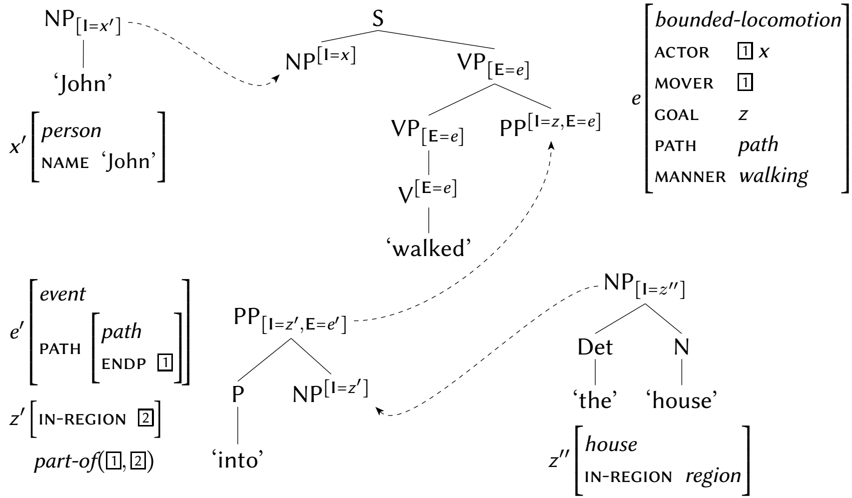
  Note: Regularities about **argument linking** are expressed in the metagrammar. [Kallmeyer/Lichte/Osswald/Petitjean 2016]

- Semantic **composition** ≈ frame unification via identification of index variables during substitution and adjunction.

LTAG & frame semantics

Example (intransitive) directed motion construction

(4) John walked into the house.



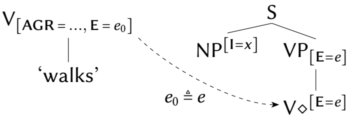
Lexical anchoring

morph entry +
'walks'
pos: V
Syn_1:
$\left[\text{AGR} = \begin{bmatrix}\text{PERS} = 3 \\ \text{NUM} = sg\end{bmatrix}\right]$
lemma: walk

lemma entry +
walk:
FAM: n0V, …
Syn_2:
$\left[\text{E} = e_0\right]$
Sem:
$e_0 \begin{bmatrix} \textit{locomotion} & \\ \text{MANNER} & \text{walking}\end{bmatrix}$

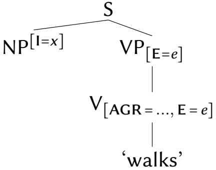
Constraints:
$\textit{locomotion} \Rightarrow \textit{activity} \wedge \textit{translocation}$
$\textit{translocation} \Rightarrow \textit{motion} \wedge \text{PATH} : \textit{path}$
$\textit{activity} \Rightarrow \text{ACTOR} : \top$
$\textit{motion} \Rightarrow \text{MOVER} : \top$
$\textit{activity} \wedge \textit{motion} \Rightarrow \text{ACTOR} \doteq \text{MOVER}$

$\rightsquigarrow \quad e_0 \begin{bmatrix} \textit{locomotion} & \\ \text{ACTOR} & \boxed{1} \\ \text{MOVER} & \boxed{1} \\ \text{PATH} & \textit{path} \\ \text{manner} & \text{walking}\end{bmatrix}$



$e \begin{bmatrix} \textit{activity} & \\ \text{ACTOR} & x\end{bmatrix}$

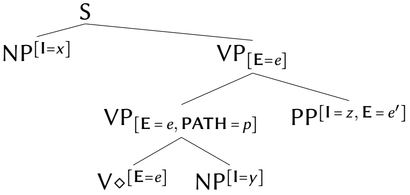
$\rightsquigarrow \quad e \begin{bmatrix} \textit{locomotion} & \\ \text{ACTOR} & \boxed{1}\ x \\ \text{MOVER} & \boxed{1} \\ \text{PATH} & \textit{path} \\ \text{MANNER} & \textit{walking}\end{bmatrix}$



Example (transitive/causative) directed motion construction

(5) Mary threw/kicked/rolled the ball into the room.

Unanchored construction (*n0Vn1pp(dir)*):

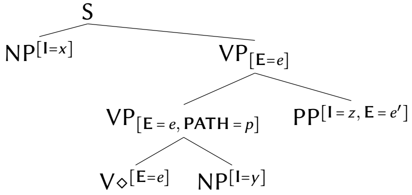


$$e\begin{bmatrix} \textit{causation} & \\ \textsc{cause} & \begin{bmatrix} \textit{activity} & \\ \textsc{actor} & x \\ \textsc{theme} & y \end{bmatrix} \\ \textsc{effect} & e'\begin{bmatrix} \textit{bounded-translocation} & \\ \textsc{mover} & y \\ \textsc{goal} & z \\ \textsc{path} & p \end{bmatrix} \end{bmatrix}$$

Example (transitive/causative) directed motion construction

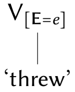
(5) Mary threw/kicked/rolled the ball into the room.

Unanchored construction (*n0Vn1pp(dir)*):



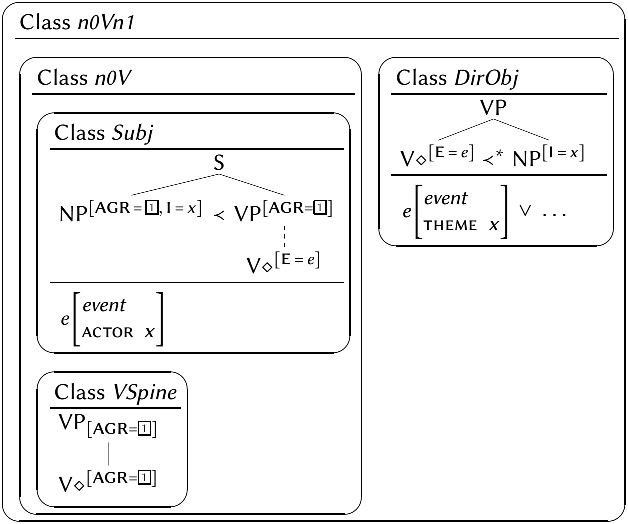
$$e\begin{bmatrix} \textit{causation} & \\ \text{CAUSE} & \begin{bmatrix}\textit{activity} & \\ \text{ACTOR} & x\\ \text{THEME} & y\end{bmatrix}\\ \text{EFFECT} & e'\begin{bmatrix}\textit{bounded-translocation} & \\ \text{MOVER} & y \\ \text{GOAL} & z\\ \text{PATH} & p\end{bmatrix}\end{bmatrix}$$

(Partial) lexical entry for ‘threw’:



$$e\begin{bmatrix} \textit{onset-causation} & \\ \text{CAUSE} & \begin{bmatrix}\textit{activity} & \\ \text{ACTOR} & \top\\ \text{THEME} & \boxed{1}\\ \text{MANNER} & \textit{throwing}\end{bmatrix}\\ \text{EFFECT} & \begin{bmatrix}\textit{translocation} \wedge \textit{undergoing} & \\ \text{THEME} & \boxed{1} \\ \text{MOVER} & \boxed{1}\end{bmatrix}\end{bmatrix}$$

Metagrammar classes (syntax + semantics)

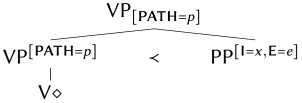


LTAG & frame semantics

Metagrammar classes (syntax + semantics)

Class *DirPrepObj*

export: e, x



$e\begin{bmatrix} \textit{bounded-translocation} & \\ \text{GOAL} & x \\ \text{PATH} & p \end{bmatrix}$

Class *n0Vpp(dir)*

identities: $C_1.e = C_2.e$

- Class C_1 =*n0V*
  - export: e
  - …
- Class C_2 =*DirPrepObj*
  - export: e, z
  - …

Class *n0Vn1pp(dir)*

identities: $C_1.e = e$, $C_2.x = z$, $C_2.e = e'$

- Class C_1 =*n0Vn1*
  - export: e
  - …
- Class C_2 =*DirPrepObj*
  - export: e, x
  - …

$$e\begin{bmatrix} \textit{causation} & \\ \text{ACTOR} & x\ \boxed{1} \\ \text{THEME} & y\ \boxed{2} \\ \text{CAUSE} & \begin{bmatrix} \textit{activity} & \\ \text{ACTOR} & \boxed{1} \\ \text{THEME} & \boxed{2} \end{bmatrix} \\ \text{EFFECT} & e'\begin{bmatrix} \text{MOVER} & \boxed{2} \\ \text{GOAL} & z \end{bmatrix} \end{bmatrix}$$

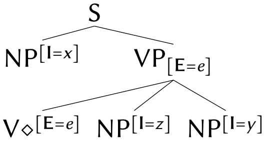
The Dative alternation (sketch)

(6) a. John sent Mary the book. [double object construction]
b. John sent the book to Mary. [prepositional object construction]

The Dative alternation (sketch)

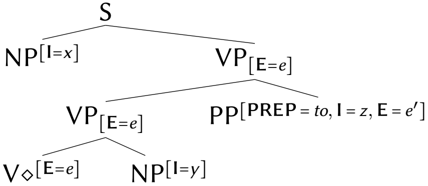
(6) a. John sent Mary the book. [double object construction]
b. John sent the book to Mary. [prepositional object construction]

a)



$$e\begin{bmatrix} \textit{causation} & \\ \text{CAUSE} & \begin{bmatrix} \textit{activity} & \\ \text{ACTOR} & x \end{bmatrix} \\ \text{EFFECT} & \begin{bmatrix} \textit{change-of-possession} & \\ \text{THEME} & y \\ \text{RECIPIENT} & z \end{bmatrix} \end{bmatrix}$$

b)

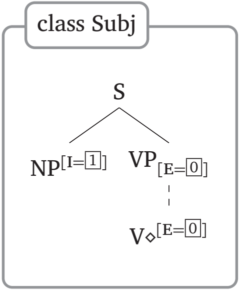


$$e\begin{bmatrix} \textit{causation} & \\ \text{CAUSE} & \begin{bmatrix} \textit{activity} & \\ \text{ACTOR} & x \end{bmatrix} \\ \text{EFFECT} & e'\begin{bmatrix} \textit{bounded-translocation} & \\ \text{MOVER} & y \\ \text{GOAL} & z \end{bmatrix} \end{bmatrix}$$

Examples

[from Lichte/Petitjean 2015]

<syn>-dimension of class *Subj*

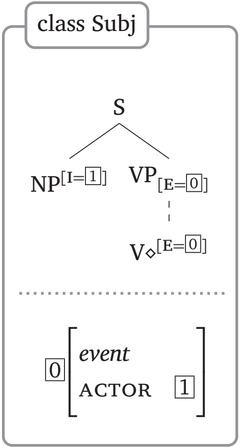



```
class Subj
...
<syn>{
  node ?S [cat=s];
  node ?SUBJ [cat=np,
              top=[i=?1]];
  node ?VP [cat=vp,bot=[e=?0]];
  node ?V (mark=anchor)
          [cat=v,top=[e=?0]];
  ?S->?SUBJ; ?S->?VP; ?VP->*?V;
  ?SUBJ>>?VP
}
```

XMG implementation of frame semantics

Examples [from Lichte/Petitjean 2015]

<syn>-dimension + <frame>-dimension of class *Subj*




```
class Subj
...
<syn>{
  node ?S [cat=s];
  node ?SUBJ [cat=np,
              top=[i=?1]];
  node ?VP [cat=vp,bot=[e=?0]];
  node ?V (mark=anchor)
          [cat=v,top=[e=?0]];
  ?S->?SUBJ; ?S->?VP; ?VP->*?V;
  ?SUBJ>>?VP
}
<frame>{
?0[event,
    actor:?1]
}
```

XMG implementation of frame semantics

Examples

[from Lichte/Petitjean 2015]

Specification of frames:

$$\boxed{0}\left[\begin{array}{ll} \textit{causation} & \\ \textsc{actor} & \boxed{1} \\ \textsc{theme} & \boxed{2} \\ \textsc{cause} & \left[\begin{array}{ll} \textit{activity} & \\ \textsc{actor} & \boxed{1} \\ \textsc{theme} & \boxed{2} \end{array}\right] \\ \textsc{effect} & \boxed{4}\left[\begin{array}{ll} \textsc{mover} & \boxed{2} \\ \textsc{goal} & \boxed{3} \end{array}\right] \end{array}\right]$$

```
<frame>{
?0[causation,
    actor:?1,
    theme:?2,
    cause:[activity,
           actor:?1,
           theme:?2],
    effect:?4[mover:?2,
              goal:?3]
]}
```

XMG implementation of frame semantics

Examples

[from Lichte/Petitjean 2015]

Specification of frames:

$$\boxed{0}\begin{bmatrix} \textit{causation} & \\ \textsc{actor} & \boxed{1} \\ \textsc{theme} & \boxed{2} \\ \textsc{cause} & \begin{bmatrix} \textit{activity} & \\ \textsc{actor} & \boxed{1} \\ \textsc{theme} & \boxed{2} \end{bmatrix} \\ \textsc{effect} & \boxed{4}\begin{bmatrix} \textsc{mover} & \boxed{2} \\ \textsc{goal} & \boxed{3} \end{bmatrix} \end{bmatrix}$$

```
<frame>{
?0[causation,
    actor:?1,
    theme:?2,
    cause:[activity,
           actor:?1,
           theme:?2],
    effect:?4[mover:?2,
              goal:?3]
]}
```

Specification of attribute-value constraints:

```
frame_constraints = {
    activity -> event, activity -> [actor:+],
    motion -> event, motion -> [mover:+],
    causation -> event, causation -> [cause:+,effect:+],
    locomotion -> activity motion}
```

Frame semantics: extensions

Obvious issue

What about sentence level semantics, quantification, intensionality, and all these things?

Frame semantics: extensions

Obvious issue

What about sentence level semantics, quantification, intensionality, and all these things?

Possible approaches

1. Use an **attribute-value language with quantifiers** (or, e.g., Hybrid Logic), and build formulas instead of models.
   [e.g., Kallmeyer/Osswald/Pogodalla 2016]

2. Keep frames as basic semantic representations and evaluate **quantification over the domain of frames**. [≈ Muskens 2013]

3. Try to retain the idea of minimal model building and consider **frame types** as proper entities of the model/universe.

Frame semantics: extensions

Attribute-value formulas with quantifiers (qAVForm)

$\forall\phi$, $\exists\phi$ ($\phi \in$ AVDesc) $\qquad$ $\boldsymbol{\forall} x\,\alpha$, $\boldsymbol{\exists} x\,\alpha$ ($\alpha \in$ AVForm $\cup$ qAVForm)

$\langle V, \mathcal{I}, g\rangle \vDash \forall\phi$ iff $\langle V, \mathcal{I}, g\rangle, v \vDash \phi$ for every $v \in V$

$\langle V, \mathcal{I}, g\rangle \vDash \exists\phi$ iff $\langle V, \mathcal{I}, g\rangle, v \vDash \phi$ for some $v \in V$

For $x \notin \mathrm{dom}(g)$:

$\langle V, \mathcal{I}, g\rangle \vDash \boldsymbol{\forall} x\,\alpha$ iff $\langle V, \mathcal{I}, g'\rangle \vDash \alpha$ for every assignment g' with
$\mathrm{dom}(g') = \mathrm{dom}(g) \cup \{x\}$ and
$g(v) = g'(v)$ for all $v \in \mathrm{dom}(g)$

$\langle V, \mathcal{I}, g\rangle \vDash \boldsymbol{\exists} x\,\alpha$ iff $\langle V, \mathcal{I}, g'\rangle \vDash \alpha$ for some assignment g' with …

Note: $\forall\phi \equiv \boldsymbol{\forall} x(x \cdot \phi)$, $\exists\phi \equiv \boldsymbol{\exists} x(x \cdot \phi)$ (with x not occurring in ϕ)

Frame semantics: extensions

Example

(7) Every man walked into some house.

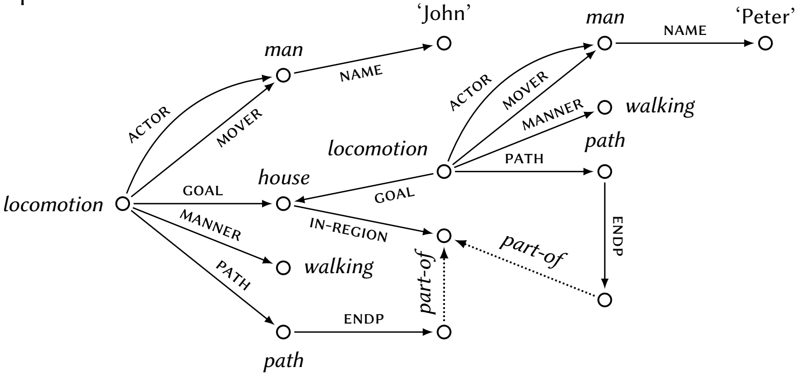
$$\forall x(x \cdot \mathit{man} \rightarrow \exists z(z \cdot \mathit{house} \wedge \exists(\mathit{locomotion} \wedge \textsc{manner} : \mathit{walking} \\ \wedge \textsc{actor} \triangleq x \wedge \textsc{mover} \doteq \textsc{actor} \\ \wedge \textsc{goal} \triangleq z \wedge \textsc{path} : (\mathit{path} \wedge \textsc{endp} : \mathit{region}) \\ \wedge [\textsc{path endp}, \textsc{goal in-region}] : \mathit{part\text{-}of})))$$

Example

(7) Every man walked into some house.

$$\forall x(x \cdot man \rightarrow \exists z(z \cdot house \wedge \exists(locomotion \wedge \text{MANNER} : walking$$
$$\wedge\ \text{ACTOR} \triangleq x \ \wedge\ \text{MOVER} \doteq \text{ACTOR}$$
$$\wedge\ \text{GOAL} \triangleq z \ \wedge\ \text{PATH} : (path \wedge \text{ENDP} : region)$$
$$\wedge\ [\text{PATH ENDP},\ \text{GOAL IN-REGION}] : part\text{-}of)))$$

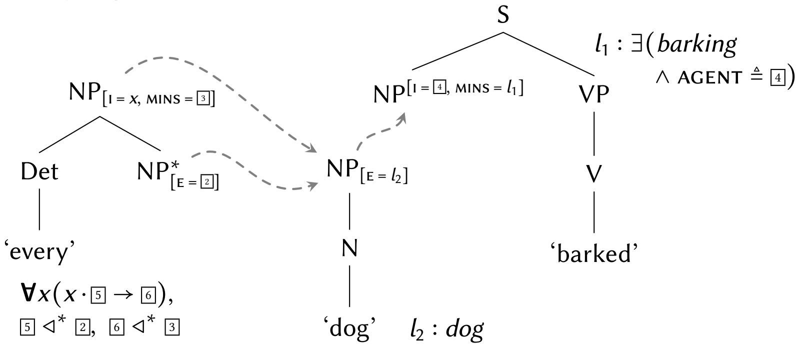
Example model



Frame semantics: extensions

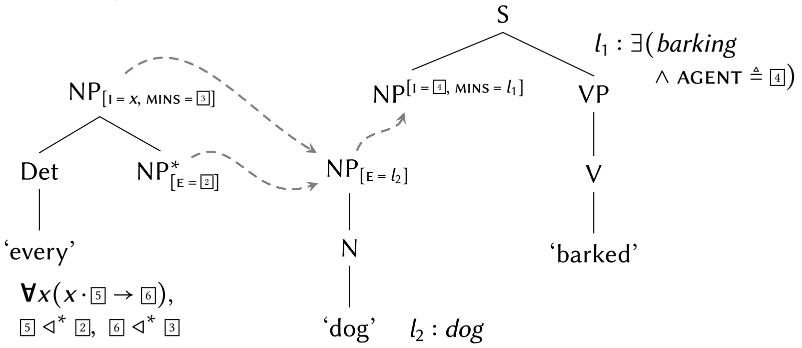
AV logic with quantifiers + underspecification (“hole semantics”)

(8) Every dog barked.



AV logic with quantifiers + underspecification ("hole semantics")

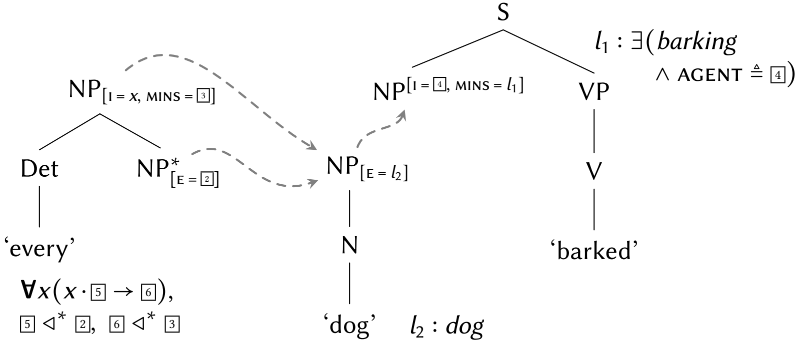
(8) Every dog barked.



$\rightsquigarrow\ \forall x(x \cdot \boxed{5} \rightarrow \boxed{6}),\ l_2 : dog,\ l_1 : \exists(barking \wedge \textsc{agent} \triangleq x),\ \boxed{5} \lhd^{*} l_2,\ \boxed{6} \lhd^{*} l_1$

AV logic with quantifiers + underspecification ("hole semantics")

(8) Every dog barked.



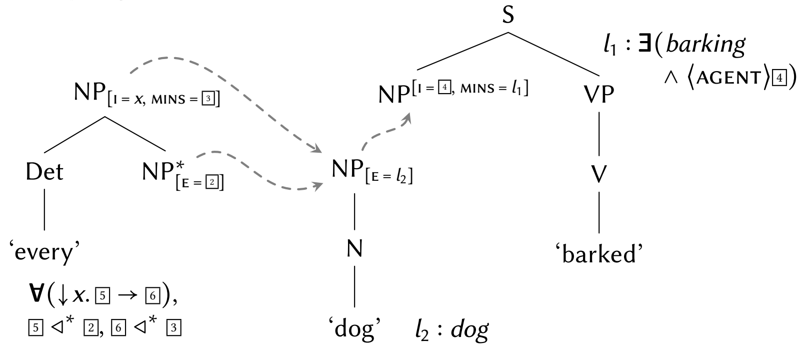
$\rightsquigarrow\ \forall x(x \cdot \boxed{5} \rightarrow \boxed{6}),\ l_2 : \textit{dog},\ l_1 : \exists(\textit{barking} \wedge \text{AGENT} \triangleq x),\ \boxed{5} \lhd^{*} l_2,\ \boxed{6} \lhd^{*} l_1$

$\rightsquigarrow\ \forall x(x \cdot \textit{dog} \rightarrow \exists(\textit{barking} \wedge \text{AGENT} \triangleq x))$

Frame semantics: extensions

Alternative: **Hybrid Logic + underspecification** (“hole semantics”)

(9) Every dog barked.



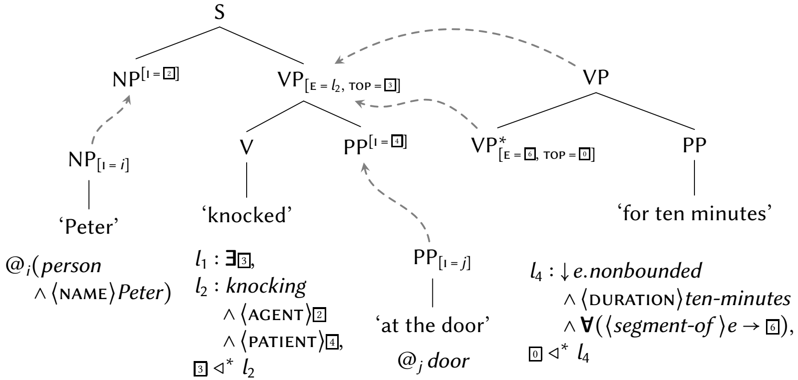
$$\rightsquigarrow\ \forall(\downarrow x. \boxed{5} \rightarrow \boxed{6}),\ l_2 : dog,\ l_1 : \exists(barking \wedge \langle\text{AGENT}\rangle x),\ \boxed{5} \lhd^{*} l_2, \boxed{6} \lhd^{*} l_1$$

$$\rightsquigarrow\ \forall(\downarrow x.\ dog \rightarrow \exists(barking \wedge \langle\text{AGENT}\rangle x))$$

[from Kallmeyer/Osswald/Pogodalla 2016]

Alternative: **Hybrid Logic + underspecification** ("hole semantics")

(10) Peter knocked at the door for ten minutes.



$$\rightsquigarrow \quad \exists(\downarrow e.\mathit{nonbounded} \wedge \langle\textsc{duration}\rangle \mathit{ten\text{-}minutes} \wedge \forall(\langle \mathit{segment\text{-}of}\rangle e \rightarrow \mathit{knocking} \wedge \langle\textsc{agent}\rangle i \wedge \langle\textsc{patient}\rangle j)) \wedge @_i(\mathit{person} \wedge \langle\textsc{name}\rangle \mathit{Peter}) \wedge @_j \mathit{door}$$

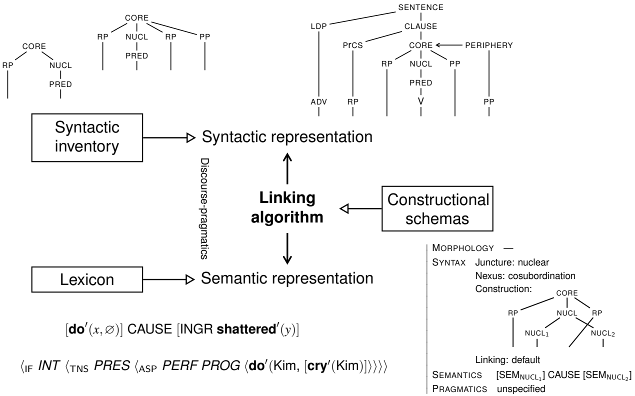
[from Kallmeyer/Osswald/Pogodalla 2016]

Further topics

Application: Formalization of Role and Reference Grammar

Role and Reference Grammar (RRG): [e.g., Van Valin 2005, 2010]

A **non-transformational** grammatical theory, inspired by typological concerns, which makes use of **syntactic templates** and **lexical decomposition** structures, among others.



Further topics

Application: Formalization of Role and Reference Grammar

Role and Reference Grammar (RRG): [e.g., Van Valin 2005, 2010]

A **non-transformational** grammatical theory, inspired by typological concerns, which makes use of **syntactic templates** and **lexical decomposition** structures, among others.

Aspects of the formalization

- Modified tree operations because of flat syntactic structures: Wrapping substitution and sister adjunction.
  [Kallmeyer/Osswald/Van Valin 2013; Osswald/Kallmeyer, to appear]
- Semantic frames instead of semantic templates.
- Argument linking rules as constraints in the metagrammar.
  [Kallmeyer/Lichte/Osswald/Petitjean 2016]

Thank you very much for your attention!

References I

Abeillé, A. and Rambow, O. (2000). Tree Adjoining Grammar: An Overview. In Abeillé, A. and Rambow, O., editors, *Tree Adjoining Grammars: Formalisms, Linguistic Analysis and Processing*, pages 1–68. CSLI.

Babonnaud, W., Kallmeyer, L., and Osswald, R. (2016). Polysemy and coercion – a frame-based approach using LTAG and Hybrid Logic. In Amblard, M., de Groote, P., Pogodalla, S., and Retoré, C., editors, *Logical Aspects of Computational Linguistics, 9th International Conference*, Lecture Notes in Artificial Intelligence 10054, pages 18–33. Springer, Berlin.

Bangalore, S. and Joshi, A. K. (2010). Introduction. In Bangalore, S. and Joshi, A. K., editors, *Supertagging: Using Complex Lexical Descriptions in Natural Language Processing*, pages 1–31. MIT Press, Cambridge, MA.

Barsalou, L. W. (1992). Frames, concepts, and conceptual fields. In Lehrer, A. and Kittay, E. F., editors, *Frames, Fields, and Contrasts*, pages 21–74. Lawrence Erlbaum Associates, Hillsdale, NJ.

Crabbé, B. and Duchier, D. (2005). Metagrammar redux. In Christiansen, H., Skadhauge, P. R., and Villadsen, J., editors, *Constraint Solving and Language Processing*, Lecture Notes in Computer Science 3438, pages 32–47. Springer, Berlin.

Crabbé, B., Duchier, D., Gardent, C., Le Roux, J., and Parmentier, Y. (2013). XMG: eXtensible MetaGrammar. *Computational Linguistics*, 39(3):591–629.

Gardent, C. and Kallmeyer, L. (2003). Semantic Construction in FTAG. In *Proceedings of EACL 2003*, pages 123–130, Budapest.

Hegner, S. J. (1994). Properties of Horn clauses in feature-structure logic. In Rupp, C. J., Rosner, M. A., and Johnson, R. L., editors, *Constraints, Language and Computation*, pages 111–147. Academic Press, San Diego, CA.

Joshi, A. K. and Schabes, Y. (1997). Tree-Adjoining Grammars. In Rozenberg, G. and Salomaa, A., editors, *Handbook of Formal Languages*, pages 69–123. Springer, Berlin.

Kallmeyer, L. and Joshi, A. K. (2003). Factoring Predicate Argument and Scope Semantics: Underspecified Semantics with LTAG. *Research on Language and Computation*, 1(1–2):3–58.

Kallmeyer, L., Lichte, T., Osswald, R., and Petitjean, S. (2016a). Argument linking in LTAG: A constraint-based implementation with XMG. In *Proceedings of the 12th International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+12)*, pages 48–57.

Kallmeyer, L. and Osswald, R. (2013). Syntax-driven semantic frame composition in Lexicalized Tree Adjoining Grammars. *Journal of Language Modelling*, 1(2):267–330.

References II

Kallmeyer, L., Osswald, R., and Pogodalla, S. (2016b). *For*-adverbials and aspectual interpretation: An LTAG analysis using hybrid logic and frame semantics. In Piñón, C., editor, *Empirical Issues in Syntax and Semantics EISS*, volume 11.

Kallmeyer, L., Osswald, R., and Van Valin, Jr., R. D. (2013). Tree wrapping for Role and Reference Grammar. In Morrill, G. and Nederhof, M.-J., editors, *Formal Grammar (FG 2012/2013)*, number 8036 in Lecture Notes in Computer Science, pages 175–190. Springer.

Kallmeyer, L. and Romero, M. (2008). Scope and situation binding in LTAG using semantic unification. *Research on Language and Computation*, 6(1):3–52.

Lichte, T. and Petitjean, S. (2015). Implementing semantic frames as typed feature structures with XMG. *Journal of Language Modelling*, 3(1):185–228.

Löbner, S. (2014). Evidence for frames from human language. In Gamerschlag, T., Gerland, D., Osswald, R., and Petersen, W., editors, *Frames and Concept Types*, number 94 in Studies in Linguistics and Philosophy, pages 23–67. Springer, Dordrecht.

Muskens, R. (2013). Data semantics and linguistic semantics. In Aloni, M., Franke, M., and Roelofsen, F., editors, *The dynamic, inquisitive, and visionary life of ϕ, ?ϕ, and $\diamond\phi$. A festschrift for Jeroen Groenendijk, Martin Stokhof, and Frank Veltman*, pages 175–183.

Nesson, R. and Shieber, S. M. (2006). Simpler TAG semantics through synchronization. In *Proceedings of the 11th Conference on Formal Grammar*, Malaga, Spain.

Nesson, R. and Shieber, S. M. (2008). Synchronous vector tag for syntax and semantics: Control verbs, relative clauses, and inverse linking. In *Proceedings of the Ninth International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+ 9)*, Tübingen, Germany.

Osswald, R. (1999). Semantics for attribute-value theories. In Dekker, P., editor, *Proceedings of the Twelfth Amsterdam Colloquium*, pages 199–204, Amsterdam. University of Amsterdam, ILLC.

Osswald, R. and Van Valin, Jr., R. D. (2014). FrameNet, frame structure, and the syntax-semantics interface. In Gamerschlag, T., Gerland, D., Osswald, R., and Petersen, W., editors, *Frames and Concept Types*, number 94 in Studies in Linguistics and Philosophy, pages 125–156. Springer, Dordrecht.

Rappaport Hovav, M. and Levin, B. (1998). Building verb meanings. In Butt, M. and Geuder, W., editors, *The Projection of Arguments: Lexical and Compositional Factors*, pages 97–134. CSLI Publications, Stanford, CA.

Shieber, S. M. (1994). Restricting the weak-generative capacity of synchronous Tree-Adjoining Grammars. *Computational Intelligence*, 10(4):271–385.

Vijay-Shanker, K. and Joshi, A. K. (1988). Feature structures based tree adjoining grammar. In *Proceedings of COLING*, pages 714–719, Budapest.