

Stacked Supplements

Daniel Gutzmann

Graduiertenkolleg »Satzarten«
Goethe-Universität Frankfurt

ZAS, Berlin – September 3, 2009

1 Introduction

2 Conventional implicatures

3 Supplements as CIs

4 Stacked supplements

Introduction

Supplements

The descriptive term *supplement* refers to a class of expressions that includes different kinds of unintegrated material that despite being embedded nevertheless receive a root-level interpretation.

Kinds of supplements

(Potts 2005: 90)

- (1) a. Ames, **who was a successful spy**, is now behind bars.
(*supplementary relative*)
- b. Ames was, **as the press reported**, a successful spy.
(*as-parenthetical*)
- c. Ames, **a successful spy**, is now behind bars.
(*nominal appositive*)

Syntax vs. semantics

- Traditionally, most accounts of supplements are *syntactic* insofar as they trace back the root status of supplements back to a special kind of syntactic structure.

Syntax vs. semantics

- Traditionally, most accounts of supplements are *syntactic* insofar as they trace back the root status of supplements back to a special kind of syntactic structure.
- In contrast to this, Potts (2002, 2005) argue for a purely semantic approach.

Syntax vs. semantics

- Traditionally, most accounts of supplements are *syntactic* insofar as they trace back the root status of supplements back to a special kind of syntactic structure.
- In contrast to this, Potts (2002, 2005) argue for a purely semantic approach.
- He bases his analysis on his understanding of the notion of *conventional implicature* (CI).

Potts' (2005) Proposal

»The analysis I offer permits us to interpret surface structures in which supplements are syntactically embedded like regular modifiers. The fact that nothing scopes over their meaning is handled in the meaning language.«

(Potts 2005: 16)

Potts' (2005) Proposal

»The analysis I offer permits us to interpret surface structures in which supplements are syntactically embedded like regular modifiers. The fact that nothing scopes over their meaning is handled in the meaning language.«
(Potts 2005: 16)

»The proposal I defend is that supplements determine routine modifier structures [...]; the distinction between at-issue and CI content is a semantic one« (Potts 2005: 90)

Argumentation of this talk

- Even if Potts' semantic approach can deal with many kinds of supplements, I show that his approach has problems with *some* special kinds of *stacked supplements*.
- To account for these cases, the semantic type system of Potts' (2005) has to be enriched.

Argumentation of this talk

- Even if Potts' semantic approach can deal with many kinds of supplements, I show that his approach has problems with *some* special kinds of *stacked supplements*.
- To account for these cases, the semantic type system of Potts' (2005) has to be enriched.
- However, this leaves some of the properties of supplements unexplained.

Argumentation of this talk

- Even if Potts' semantic approach can deal with many kinds of supplements, I show that his approach has problems with *some* special kinds of *stacked supplements*.
- To account for these cases, the semantic type system of Potts' (2005) has to be enriched.
- However, this leaves some of the properties of supplements unexplained.
- Hence, it seems that we are still in need of a syntactic explanation of the unexplained facts about supplements.

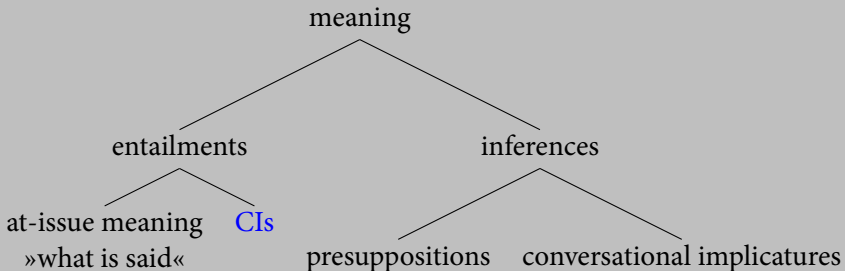
Conventional implicatures

- In his 2005 book, Potts develops a fresh view on the Gricean notion of conventional implicature (CI).
- In contrast to Grice's original characterization, he thinks of CIs as being a semantic kind of meaning rather than a pragmatic one.

- In his 2005 book, Potts develops a fresh view on the Gricean notion of conventional implicature (CI).
- In contrast to Grice's original characterization, he thinks of CIs as being a semantic kind of meaning rather than a pragmatic one.

Kinds of meaning

(2)



Main properties of CIs

(Potts 2005: 11)

- (3)
- a. CIs are part of the lexical meaning of words.
 - b. CIs are commitments, and thus give rise to entailments.
 - c. These commitments are made by *the speaker of the utterance* ›by virtue of the meaning of the‹ words he chooses.
 - d. CIs are logically and compositionally independent of what is »said (in the favoured sense)«, i.e. independent of the at-issue entailments.

Main properties of CIs

(Potts 2005: 11)

- (3)
- CIs are part of the lexical meaning of words.
 - CIs are commitments, and thus give rise to entailments.
 - These commitments are made by *the speaker of the utterance* ›by virtue of the meaning of the‹ words he chooses.
 - CIs are logically and compositionally independent of what is »said (in the favoured sense)«, i.e. independent of the at-issue entailments.

Derived properties of CIs

(Potts 2005: 41-44)

- (4)
- CIs are scopeless (take widest scope).
 - CIs are not context dependent.
 - CIs yield multidimensional content.
 - ...

Formal semantics for CIs

- Potts uses a multidimensional logic \mathfrak{L}_{CI} to analyse expressions that convey CIs.
- The logic \mathfrak{L}_{CI} extends a standard type-driven, model-theoretic logic by the introduction of ...

Formal semantics for CIs

- Potts uses a multidimensional logic \mathfrak{L}_{CI} to analyse expressions that convey CIs.
 - The logic \mathfrak{L}_{CI} extends a standard type-driven, model-theoretic logic by the introduction of ...
- ① A new basic type t^c for CI-propositions

Formal semantics for CIs

- Potts uses a multidimensional logic \mathfrak{L}_{CI} to analyse expressions that convey CIs.
- The logic \mathfrak{L}_{CI} extends a standard type-driven, model-theoretic logic by the introduction of ...
 - ① A new basic type t^c for CI-propositions
 - ② Rules to construct complex CI-types

Formal semantics for CIs

- Potts uses a multidimensional logic \mathfrak{L}_{CI} to analyse expressions that convey CIs.
- The logic \mathfrak{L}_{CI} extends a standard type-driven, model-theoretic logic by the introduction of ...
 - ① A new basic type t^c for CI-propositions
 - ② Rules to construct complex CI-types
 - ③ Tree-admissibility conditions that regulate how CI-expressions compose with other expressions

Formal semantics for CIs

- Potts uses a multidimensional logic \mathfrak{L}_{CI} to analyse expressions that convey CIs.
- The logic \mathfrak{L}_{CI} extends a standard type-driven, model-theoretic logic by the introduction of ...
 - ① A new basic type t^c for CI-propositions
 - ② Rules to construct complex CI-types
 - ③ Tree-admissibility conditions that regulate how CI-expressions compose with other expressions
 - ④ A mechanism to interpret semantic parsetrees

Types for \mathcal{L}_{CI}

(simplified)

Types for \mathcal{L}_{CI}

(simplified)

- (5) a. e and t^a are at-issue types.

Types for \mathcal{L}_{CI}

(simplified)

- (5) a. e and t^a are at-issue types.
b. t^c is a CI type.

Types for \mathcal{L}_{CI}

(simplified)

- (5)
- a. e and t^a are at-issue types.
 - b. t^c is a CI type.
 - c. If σ and τ are at-issue types, then $\langle \sigma, \tau \rangle$ is an at-issue type.

Types for \mathcal{L}_{CI}

(simplified)

- (5)
- e and t^a are at-issue types.
 - t^c is a CI type.
 - If σ and τ are at-issue types, then $\langle \sigma, \tau \rangle$ is an at-issue type.
 - If σ is an at-issue type and τ is a CI type, then $\langle \sigma, \tau \rangle$ is a CI type.

Types for \mathcal{L}_{CI}

(simplified)

- (5)
- a. e and t^a are at-issue types.
 - b. t^c is a CI type.
 - c. If σ and τ are at-issue types, then $\langle \sigma, \tau \rangle$ is an at-issue type.
 - d. If σ is an at-issue type and τ is a CI type, then $\langle \sigma, \tau \rangle$ is a CI type.
 - e. The set of types is the union of all at-issue and CI types.

Types for \mathcal{L}_{CI}

(simplified)

- (5)
- e and t^a are at-issue types.
 - t^c is a CI type.
 - If σ and τ are at-issue types, then $\langle \sigma, \tau \rangle$ is an at-issue type.
 - If σ is an at-issue type and τ is a CI type, then $\langle \sigma, \tau \rangle$ is a CI type.
 - The set of types is the union of all at-issue and CI types.

Denotations for CIs

- The domain of type t^c is $\{1, 0\} = D_t$.
- Conventional implicatures receive a truth-conditional interpretation just like ordinary at-issue propositions.

CIs and semantic composition

- What is special about CI-expressions is the way they compose with other expressions.

CIs and semantic composition

- What is special about CI-expressions is the way they compose with other expressions.
- They do not have any influence on the at-issue content.

CIs and semantic composition

- What is special about CI-expressions is the way they compose with other expressions.
- They do not have any influence on the at-issue content.
- Instead, they express an independent proposition.

CIs and semantic composition

- What is special about CI-expressions is the way they compose with other expressions.
- They do not have any influence on the at-issue content.
- Instead, they express an independent proposition.

Independence

(6) Ames, who was a successful spy, is now behind bars.

CIs and semantic composition

- What is special about CI-expressions is the way they compose with other expressions.
- They do not have any influence on the at-issue content.
- Instead, they express an independent proposition.

Independence

- (6) Ames, who was a successful spy, is now behind bars.
- a. *at-issue*: Ames is now behind bars.

CIs and semantic composition

- What is special about CI-expressions is the way they compose with other expressions.
- They do not have any influence on the at-issue content.
- Instead, they express an independent proposition.

Independence

- (6) Ames, who was a successful spy, is now behind bars.
- a. *at-issue*: Ames is now behind bars.
 - b. *CI*: Ames was a successful spy.

Tree-admissibility conditions

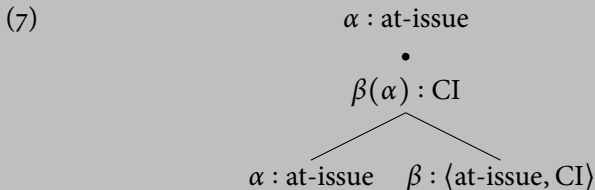
This is achieved by different tree-admissibility conditions that constrain possible semantic parsetrees.

Tree-admissibility conditions

This is achieved by different tree-admissibility conditions that constrain possible semantic parsetrees.

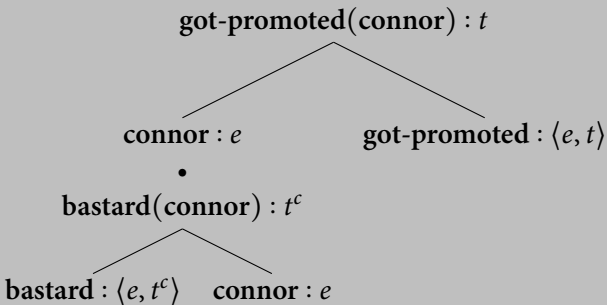
CI application

(Potts 2005: 48)



That **bastard** Connor got promoted

(8)



Parsetree interpretation (PT)

- To ensure that CI-content that has been isolated by CI application gets nevertheless interpreted, Potts (2005) uses a mechanism of *parsetree interpretation* (PT).

Parsetree interpretation (PT)

- To ensure that CI-content that has been isolated by CI application gets nevertheless interpreted, Potts (2005) uses a mechanism of *parsetree interpretation* (PT).
- According to PT, the interpretation of a semantic parsetree is a tuple consisting of ...

Parsetree interpretation (PT)

- To ensure that CI-content that has been isolated by CI application gets nevertheless interpreted, Potts (2005) uses a mechanism of *parsetree interpretation* (PT).
- According to PT, the interpretation of a semantic parsetree is a tuple consisting of ...
 - ① its at-issue content as represented by the root node, and

Parsetree interpretation (PT)

- To ensure that CI-content that has been isolated by CI application gets nevertheless interpreted, Potts (2005) uses a mechanism of *parsetree interpretation* (PT).
- According to PT, the interpretation of a semantic parsetree is a tuple consisting of ...
 - ① its at-issue content as represented by the root node, and
 - ② the set of all isolated CI proposition inside the tree.

Parsetree interpretation (PT)

- To ensure that CI-content that has been isolated by CI application gets nevertheless interpreted, Potts (2005) uses a mechanism of *parsetree interpretation* (PT).
- According to PT, the interpretation of a semantic parsetree is a tuple consisting of ...
 - ① its at-issue content as represented by the root node, and
 - ② the set of all isolated CI proposition inside the tree.
- This mechanism ensures that CI propositions receive a root-level interpretation.

That **bastard** Connor got promoted

(9) $\text{got-promoted}(\text{connor}) : t$

$\text{connor} : e$

$\text{got-promoted} : \langle e, t \rangle$

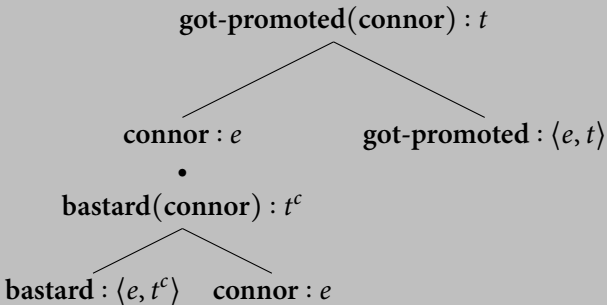
•

$\text{bastard}(\text{connor}) : t^c$

$\text{bastard} : \langle e, t^c \rangle$ $\text{connor} : e$

That **bastard** Connor got promoted

(9)

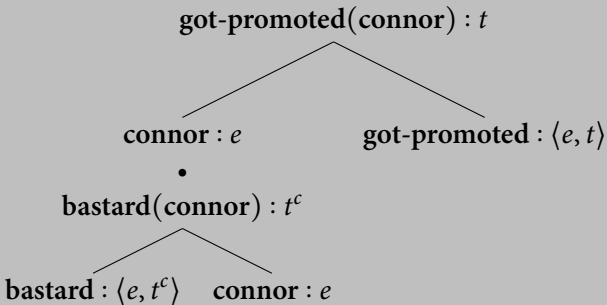


PT for »That **bastard** Connor got promoted«

(10) $\llbracket S \rrbracket = \langle \llbracket \text{at-issue content} \rrbracket, \llbracket \text{CI}_1 \rrbracket \dots \llbracket \text{CI}_n \rrbracket \rangle$

That **bastard** Connor got promoted

(9)

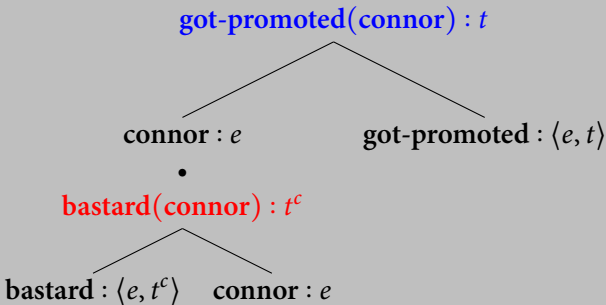


PT for »That **bastard** Connor got promoted«

(10) $\llbracket (9) \rrbracket = \langle \llbracket \text{at-issue content} \rrbracket, \llbracket \text{CI} \rrbracket \rangle$

That **bastard** Connor got promoted

(9)

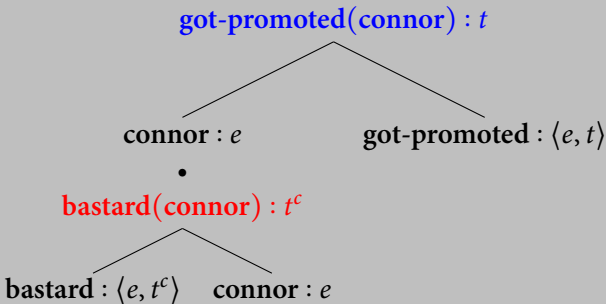


PT for »That **bastard** Connor got promoted«

(10) $\llbracket (9) \rrbracket = \langle \llbracket \text{at-issue content} \rrbracket, \llbracket \text{CI} \rrbracket \rangle$

That **bastard** Connor got promoted

(9)



PT for »That **bastard** Connor got promoted«

(10) $\llbracket (9) \rrbracket = \langle \llbracket \text{got-promoted(Connor)} \rrbracket, \llbracket \text{bastard(Connor)} \rrbracket \rangle$

Supplements as CIs

A CI analysis of supplements

- Potts uses the tools of \mathcal{L}_{CI} to analyse supplements as CI-expressions.

A CI analysis of supplements

- Potts uses the tools of \mathcal{L}_{CI} to analyse supplements as CI-expressions.

»A virtue of using the CI logic \mathcal{L}_{CI} in an analysis of supplements is that it affords a surface-true analysis: syntactically, supplements can appear as regular modifiers. Their CI-based meaning (terms of type τ^c) separate them from surrounding content, thereby providing formalization of the intuition that they represent a distinct dimension of meaning.« (Potts 2005: 97)

COMMA feature

- A crucial role in Potts' analysis of supplements as being CIs is played by a syntactic COMMA feature.

COMMA feature

- A crucial role in Potts' analysis of supplements as being CIs is played by a syntactic COMMA feature.
- This feature is represented by orthographic means (commas, parenthesis, dashes, ...) or by isolating intonation.

COMMA feature

- A crucial role in Potts' analysis of supplements as being CIs is played by a syntactic COMMA feature.
- This feature is represented by orthographic means (commas, parenthesis, dashes, ...) or by isolating intonation.
- Semantically, **comma** is an expression that takes an at-issue expression as an argument and transforms it into a CI-expression (Potts 2005: 98).

COMMA feature

- A crucial role in Potts' analysis of supplements as being CIs is played by a syntactic COMMA feature.
- This feature is represented by orthographic means (commas, parenthesis, dashes, ...) or by isolating intonation.
- Semantically, **comma** is an expression that takes an at-issue expression as an argument and transforms it into a CI-expression (Potts 2005: 98).

$$(11) \quad \text{COMMA} \rightsquigarrow \lambda f. \lambda x. f(x) : \langle \langle \sigma^a, t \rangle, \langle \sigma^a, t^c \rangle \rangle$$

- The comma feature is not represented as a distinct leaf, neither in the syntax nor in the semantics.
- In the syntactic structure, it is a feature on a node.

- The comma feature is not represented as a distinct leaf, neither in the syntax nor in the semantics.
- In the syntactic structure, it is a feature on a node.
- In the semantics it is introduced by a tree-admissibility condition for *feature semantics*.

- The comma feature is not represented as a distinct leaf, neither in the syntax nor in the semantics.
- In the syntactic structure, it is a feature on a node.
- In the semantics it is introduced by a tree-admissibility condition for *feature semantics*.

Feature semantics

$$(12) \quad \begin{array}{c} \beta(\alpha) : \tau \\ | \\ \alpha : \sigma \end{array} \quad (\text{where } \beta \text{ is a } \textit{feature term} \text{ of type } \langle \sigma, \tau \rangle)$$

- The comma feature is not represented as a distinct leaf, neither in the syntax nor in the semantics.
- In the syntactic structure, it is a feature on a node.
- In the semantics it is introduced by a tree-admissibility condition for *feature semantics*.

Feature semantics

$$(12) \quad \beta(\alpha) : \tau \quad (\text{where } \beta \text{ is a feature term of type } \langle \sigma, \tau \rangle)$$
$$\quad \quad \quad |$$
$$\quad \quad \quad \alpha : \sigma$$

..., a cyclist, ...

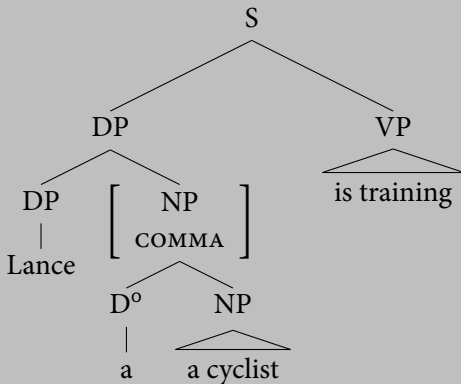
$$(13) \quad \text{comma}(\text{cyclist}) : \langle e, t^c \rangle$$
$$\quad \quad \quad |$$
$$\quad \quad \quad \text{cyclist} : \langle e, t \rangle$$

(14) Lance, a cyclist, is training.

Syntax

(Potts 2005: 97)

(15)

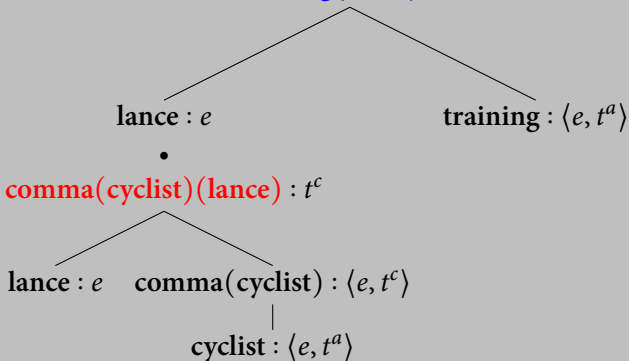


(14) Lance, a cyclist, is training.

Semantics

(Potts 2005: 97)

(15) $\text{training}(\text{lance}) : t^a$



(16) $\llbracket (15) \rrbracket = \langle \llbracket \text{training}(\text{lance}) \rrbracket, \llbracket \text{comma}(\text{cyclist})(\text{lance}) \rrbracket \rangle$

»Embedded« supplements

(Potts 2005: 117)

(17) Sheila believes that Chuck, [a psychopath](#), is fit to watch the kids.

»Embedded« supplements

(Potts 2005: 117)

- (17) Sheila believes that Chuck, [a psychopath](#), is fit to watch the kids.
- a. *at-issue*: Sheila believes that Chuck is fit to watch the kids.

»Embedded« supplements

(Potts 2005: 117)

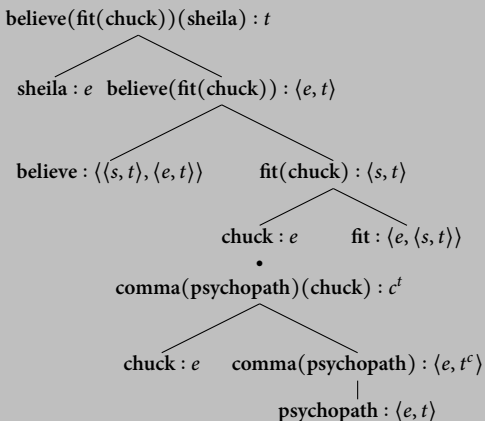
- (17) Sheila believes that Chuck, **a psychopath**, is fit to watch the kids.
- a. *at-issue*: Sheila believes that Chuck is fit to watch the kids.
 - b. *CI*: Chuck is a psychopath.

»Embedded« supplements

(Potts 2005: 117)

- (17) Sheila believes that Chuck, **a psychopath**, is fit to watch the kids.
- at-issue*: Sheila believes that Chuck is fit to watch the kids.
 - CI*: Chuck is a psychopath.

(18)



Stacked supplements

Stacked supplements

- I use the term *stacked supplements* to refer to different kinds of supplements occurring together.

Stacked supplements

- I use the term *stacked supplements* to refer to different kinds of supplements occurring together.

- (19)
- Ames, a successful spy, a former linguist, is now behind bars.
 - Ames, a successful spy, a dangerous profession, is now behind bars.
 - Ames, a successful spy, as the press reported, is now behind bars.

Type 1: Stacked on the same anchor

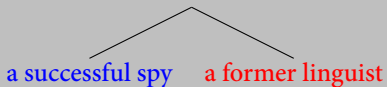
Type 1: Stacked on the same anchor

- (20) a. Ames, a successful spy, a former linguist, is now behind bars.

Type 1: Stacked on the same anchor

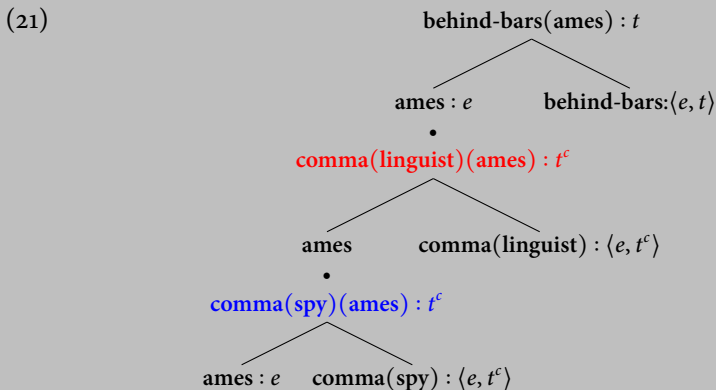
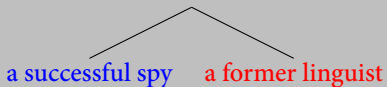
(20) a. Ames, a successful spy, a former linguist, is now behind bars.

b. Ames is now behind bars.



Type 1: Stacked on the same anchor

- (20) a. Ames, a successful spy, a former linguist, is now behind bars.
 b. Ames is now behind bars.



Type 2: Anchor inside the first supplement

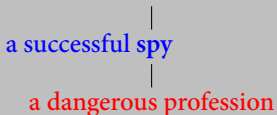
Type 2: Anchor inside the first supplement

- (22) a. Ames, a successful spy, a dangerous profession, is now behind bars.

Type 2: Anchor inside the first supplement

(22) a. Ames, a successful spy, a dangerous profession, is now behind bars.

b. Ames is now behind bars.



Type 3: First supplement is the anchor for the second one

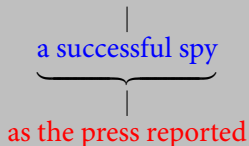
Type 3: First supplement is the anchor for the second one

- (24) a. Ames, a successful spy, as the press reported, is now behind bars.

Type 3: First supplement is the anchor for the second one

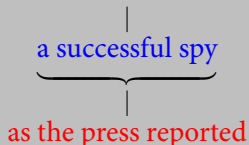
(24) a. Ames, a successful spy, as the press reported, is now behind bars.

b. Ames is now behind bars.

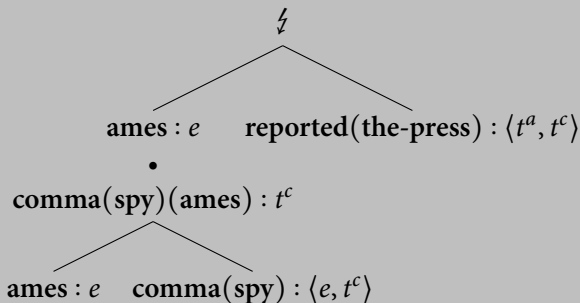


Type 3: First supplement is the anchor for the second one

- (24) a. Ames, a successful spy, as the press reported, is now behind bars.
 b. Ames is now behind bars.



(25)



- To treat the second supplement as a modifier of the first one and combine them first to yield the property of »being a successful spy as the press reported«, is not an option either, because

- To treat the second supplement as a modifier of the first one and combine them first to yield the property of »being a successful spy as the press reported«, is not an option either, because
- in order to do so, the second supplement has to be of type $\langle\langle e, t^c \rangle, \langle e, t^c \rangle\rangle$.

- To treat the second supplement as a modifier of the first one and combine them first to yield the property of »being a successful spy as the press reported«, is not an option either, because
- in order to do so, the second supplement has to be of type $\langle\langle e, t^c \rangle, \langle e, t^c \rangle\rangle$.
- But this is not a type of \mathcal{L}_{CI} !

- To treat the second supplement as a modifier of the first one and combine them first to yield the property of »being a successful spy as the press reported«, is not an option either, because
- in order to do so, the second supplement has to be of type $\langle\langle e, t^c \rangle, \langle e, t^c \rangle\rangle$.
- But this is not a type of \mathcal{L}_{CI} !

Types for \mathcal{L}_{CI}

(simplified)

- (26)
- e and t^a are at-issue types.
 - t^c is a CI type.
 - If σ and τ are at-issue types, then $\langle\sigma, \tau\rangle$ is an at-issue type.
 - If σ is an at-issue type and τ is a CI type, then $\langle\sigma, \tau\rangle$ is a CI type.
 - The set of types is the union of all at-issue and CI types.

- To account for these kinds of supplements by the tools of \mathcal{L}_{CI} , the semantic type system as to be extended.

Extended types for \mathcal{L}_{CI+}

- (27) a. e and t are at-issue types.

- To account for these kinds of supplements by the tools of \mathcal{L}_{CI} , the semantic type system as to be extended.

Extended types for \mathcal{L}_{CI+}

- (27)
- e and t are at-issue types.
 - t^c is a CI type.

- To account for these kinds of supplements by the tools of \mathcal{L}_{CI} , the semantic type system as to be extended.

Extended types for \mathcal{L}_{CI+}

- (27)
- e and t are at-issue types.
 - t^c is a CI type.
 - If σ and τ are at-issue types, then $\langle \sigma, \tau \rangle$ is an at-issue type.

- To account for these kinds of supplements by the tools of \mathfrak{L}_{CI} , the semantic type system as to be extended.

Extended types for \mathfrak{L}_{CI+}

- (27)
- e and t are at-issue types.
 - t^c is a CI type.
 - If σ and τ are at-issue types, then $\langle \sigma, \tau \rangle$ is an at-issue type.
 - If σ is an at-issue type and τ is a (hybrid or pure) CI type, then $\langle \sigma, \tau \rangle$ is a hybrid CI type.

- To account for these kinds of supplements by the tools of \mathfrak{L}_{CI} , the semantic type system as to be extended.

Extended types for \mathfrak{L}_{CI+}

- (27)
- e and t are at-issue types.
 - t^c is a CI type.
 - If σ and τ are at-issue types, then $\langle \sigma, \tau \rangle$ is an at-issue type.
 - If σ is an at-issue type and τ is a (hybrid or pure) CI type, then $\langle \sigma, \tau \rangle$ is a hybrid CI type.
 - If σ and τ are (hybrid or pure) CI types, then $\langle \sigma, \tau \rangle$ is a pure CI type.

New tree-admissibility conditions

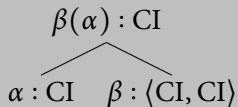
- The former tree-admissibility conditions for CI application is now divided into two different conditions.

New tree-admissibility conditions

- The former tree-admissibility conditions for CI application is now divided into two different conditions.

Pure CI application

(28)



New tree-admissibility conditions

- The former tree-admissibility conditions for CI application is now divided into two different conditions.

Pure CI application

(28)

$$\begin{array}{c} \beta(\alpha) : \text{CI} \\ \diagdown \quad \diagup \\ \alpha : \text{CI} \quad \beta : \langle \text{CI}, \text{CI} \rangle \end{array}$$

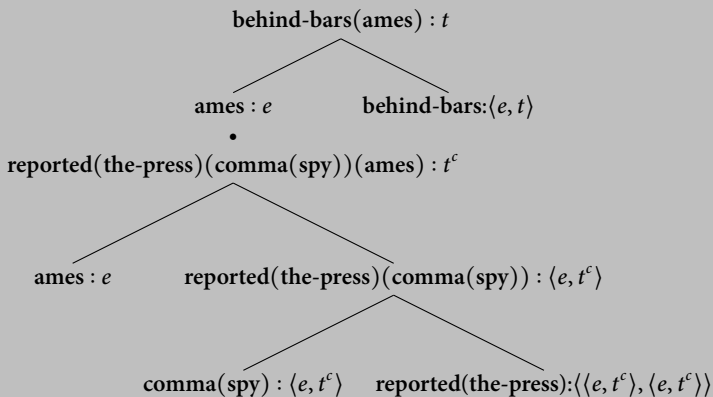
Hybrid CI application

(29)

$$\begin{array}{c} \alpha : \text{at-issue} \\ \bullet \\ \beta(\alpha) : \text{CI} \\ \diagdown \quad \diagup \\ \alpha : \text{at-issue} \quad \beta : \langle \text{at-issue}, \text{CI} \rangle \end{array}$$

Stacking resolved

(30)



Problems with this »analysis«

Problems with this »analysis«

- It is pretty *ad hoc*: why and how does *as the press reported* become an modifier on CI-expressions?

Problems with this »analysis«

- It is pretty *ad hoc*: why and how does *as the press reported* become an modifier on CI-expressions?
- Why does it become a modifier on properties, whereas it intuitively takes a propositional argument?

Problems with this »analysis«

- It is pretty *ad hoc*: why and how does *as the press reported* become an modifier on CI-expressions?
- Why does it become a modifier on properties, whereas it intuitively takes a propositional argument?
- Furthermore, allowing supplements being modified by other expression looses some of the nice generalizations of the semantics approach.

A modest conclusion

A modest conclusion

- At least for stacked supplements of type 3, a syntactic explanation seems inevitable, at least if one wants to keep the semantic approach clean.

A modest conclusion

- At least for stacked supplements of type 3, a syntactic explanation seems inevitable, at least if one wants to keep the semantic approach clean.
- However, as soon as you allow for a syntactic supplement relation for type 3 stacking, you seem to lose the virtues of a pure semantic approach.

Thank you for your attention, which I appreciate, as you may guess!