

Gapping without surprise

Toward an ellipsis-agnostic model of context dependence

Timm Lichte

Heinrich-Heine-University Düsseldorf

DGfS 2018 – AG 12

Relating Elliptical Utterances to Information in Context



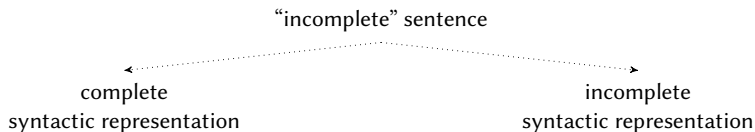
- 1 Introduction
- 2 TUCO in a nutshell
- 3 Gapping with TUCO
- 4 Gapping resolution with TUCO
- 5 Summary

- 1 Introduction
- 2 TUCO in a nutshell
- 3 Gapping with TUCO
- 4 Gapping resolution with TUCO
- 5 Summary

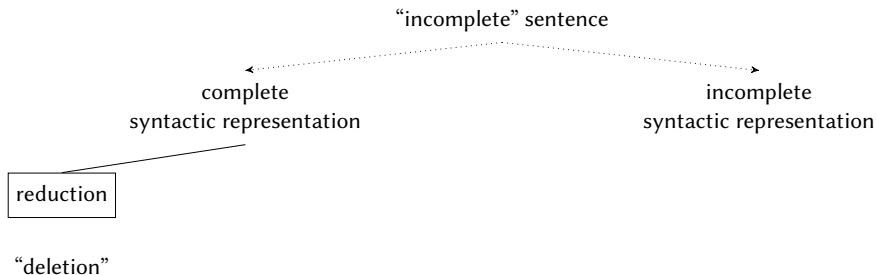
How to deal with ellipsis within a precision grammar?
(CFG, TG, MG, HPSG, CCG, TAG ...)

“incomplete” sentence

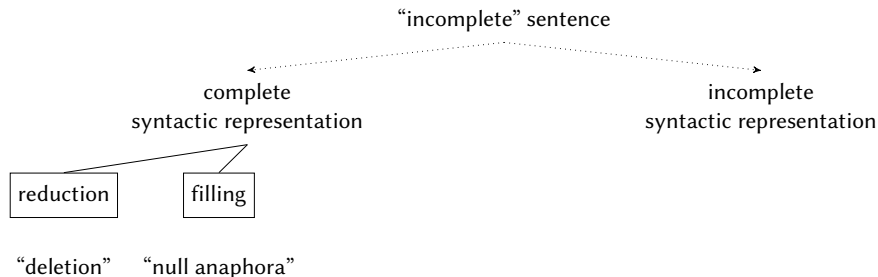
How to deal with ellipsis within a precision grammar?
(CFG, TG, MG, HPSG, CCG, TAG ...)



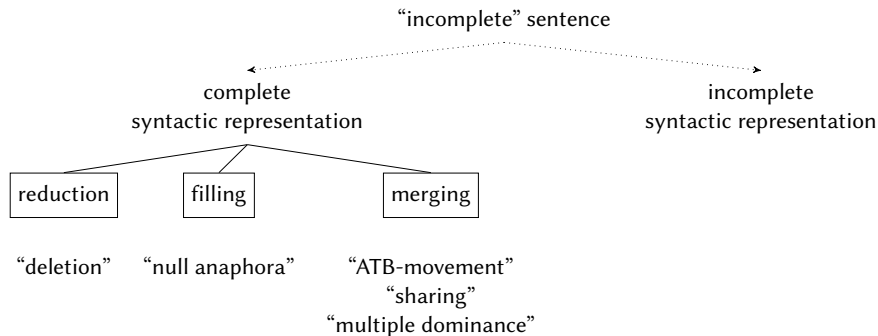
How to deal with ellipsis within a precision grammar?
(CFG, TG, MG, HPSG, CCG, TAG ...)



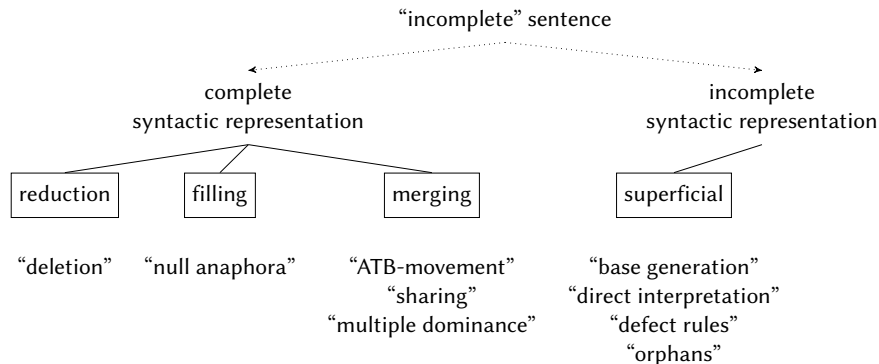
How to deal with ellipsis within a precision grammar?
(CFG, TG, MG, HPSG, CCG, TAG ...)



How to deal with ellipsis within a precision grammar?
(CFG, TG, MG, HPSG, CCG, TAG ...)



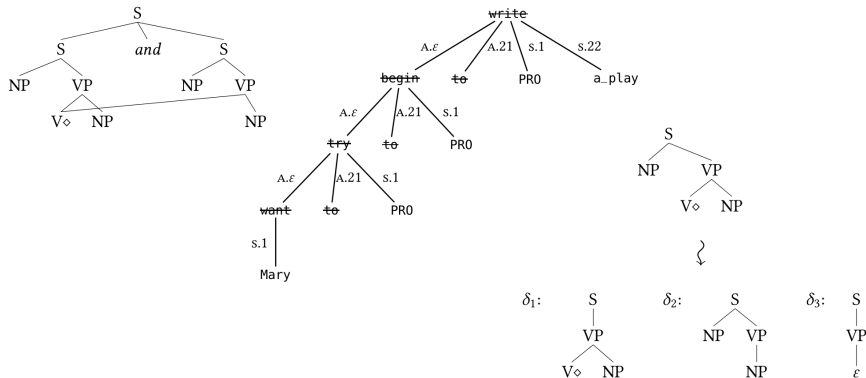
How to deal with ellipsis within a precision grammar?
(CFG, TG, MG, HPSG, CCG, TAG ...)



Background: Tree-Adjoining Grammar

Tree-Adjoining Grammar: a constructionist framework

- simple setup: elementary trees and two basic operations (substitution and adjunction)
- but: coordination and ellipsis is rather problematic (as usual)



Objectives

- 1 to model ellipsis as something syntactically “unsurprising”
 - 2 to model ellipsis resolution as recycling of syntactic fragments
- ⇒ TUCO: Tree Unification & COstraints on “derived” trees

Objectives

- 1 to model ellipsis as something syntactically “unsurprising”
 - 2 to model ellipsis resolution as recycling of syntactic fragments
- ⇒ TUCO: Tree Unification & COstraints on “derived” trees

Other goals:

- correct prediction of valid form-meaning pairs
- human-friendly representation
- not so much: performance modeling

Objectives

- 1 to model ellipsis as something syntactically “unsurprising”
 - 2 to model ellipsis resolution as recycling of syntactic fragments
- ⇒ TUCO: Tree Unification & COstraints on “derived” trees

Other goals:

- correct prediction of valid form-meaning pairs
- human-friendly representation
- not so much: performance modeling

Gapping data

- (1) a. I wrote a novel, and Mary ~~wrote~~ a play.
b. I want to try to begin to write a novel, and
Mary ~~wants to try to begin to write~~ a play.
c. I wrote a novel, and Mary ~~wrote a novel~~, too.

- 1 Introduction
- 2 TUCO in a nutshell**
- 3 Gapping with TUCO
- 4 Gapping resolution with TUCO
- 5 Summary

TUCO components:

- Tree Unification Grammar (TUG)
- (semi-)monotonic tree constraints on the derived tree

TUCO in a nutshell

TUCO components:

- Tree Unification Grammar (TUG)
- (semi-)monotonic tree constraints on the derived tree

Idea

- Syntactic structure is generated with TUG.
- The generation is guided by tree constraints.
- Tree constraints may trigger predictions.
- Tree constraints express knowledge about constructions (lexicalized or abstract).

TUCO components:

- Tree Unification Grammar (TUG)
- (semi-)monotonic tree constraints on the derived tree

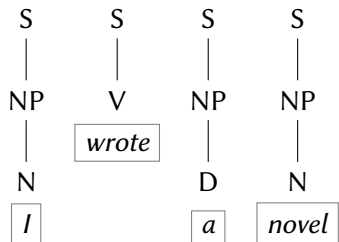
Idea

- Syntactic structure is generated with TUG.
- The generation is guided by tree constraints.
- Tree constraints may trigger predictions.
- Tree constraints express knowledge about constructions (lexicalized or abstract).

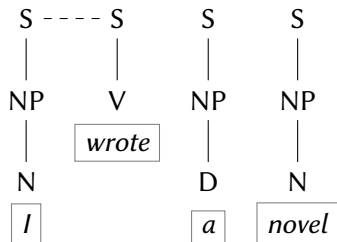
Why tree unification? It's commutative! It enables flat syntactic structures!

Why tree constraints? They're flexibly factorizable!

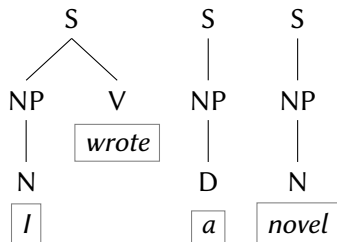
Tree unification



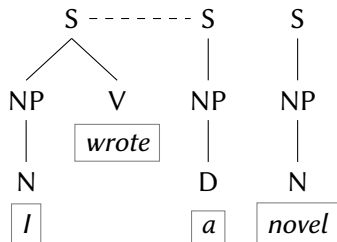
Tree unification



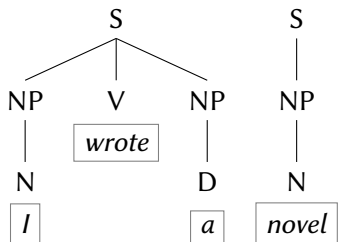
Tree unification



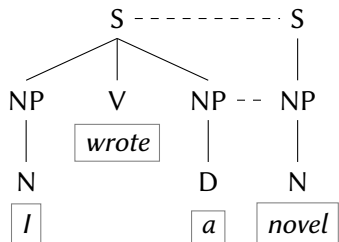
Tree unification



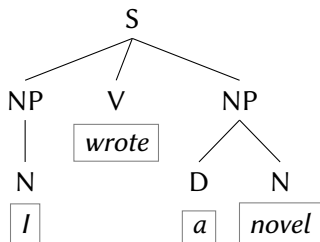
Tree unification



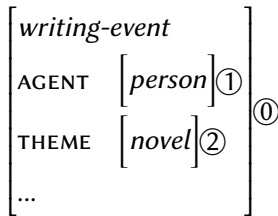
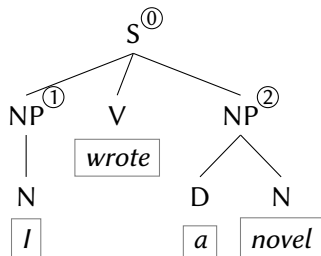
Tree unification



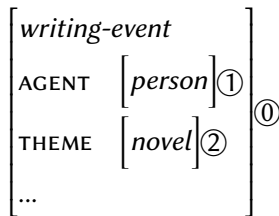
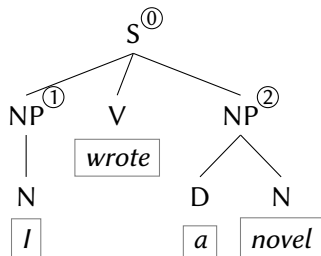
Tree unification



Tree unification



Tree unification



agreement? valency? linearization? constructional linking?

⇒ **tree constraints**

Tree constraints use **tree descriptions**:

- nodes with optional label, properties and polarization marker:
 $l!\{p_1, \dots, p_n\}$
- dominance: $\rightarrow, \rightarrow^+, \rightarrow^*$
- precedence: $<, <^+, <^*$
- conjunction and disjunction

Tree constraints

Tree constraints use **tree descriptions**:

- nodes with optional label, properties and polarization marker:
 $l!\{p_1, \dots, p_n\}$
- dominance: $\rightarrow, \rightarrow^+, \rightarrow^*$
- precedence: $<, <^+, <^*$
- conjunction and disjunction

Tree constraints

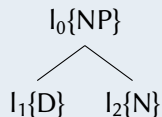
Let D_1 and D_2 be sets of descriptions, constraints have the form

$$D_1 \Rightarrow D_2$$

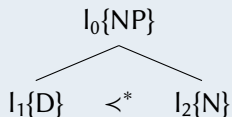
and are true in a Model \mathcal{M} , iff \mathcal{M} satisfies $D_1 \cup D_2$ or it does not satisfy D_1 .

Tree constraints: linear precedence and uniqueness

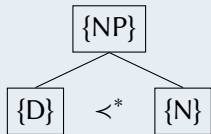
Linear precedence between determiners and nouns:



\Rightarrow

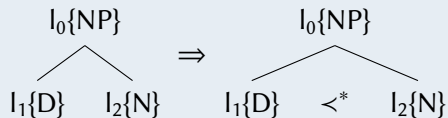


or for short:

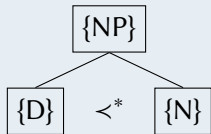


Tree constraints: linear precedence and uniqueness

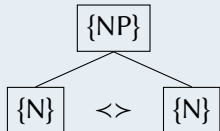
Linear precedence between determiners and nouns:



or for short:

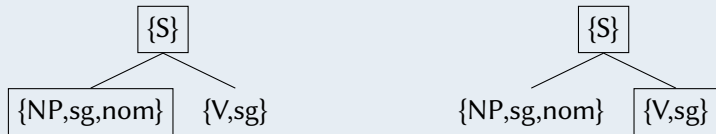


Uniqueness of the noun by contradiction:



Tree constraints: agreement

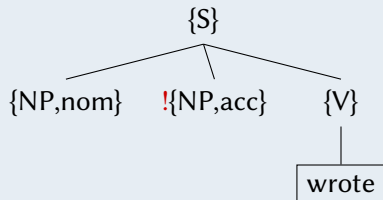
Subject-verb agreement (two constraints):



Agreement can be also dealt with in the parent node.

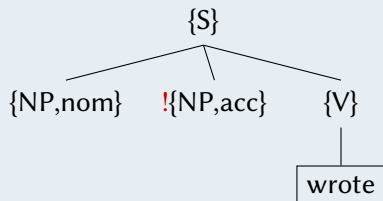
Tree constraints: valency

Valency also triggers the prediction of nodes (potentially without yield):

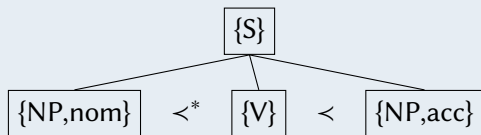


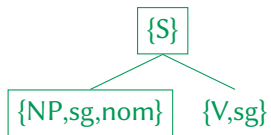
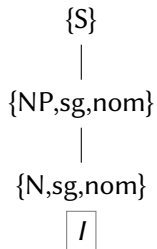
Tree constraints: valency

Valency also triggers the prediction of nodes (potentially without yield):

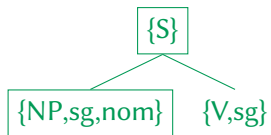
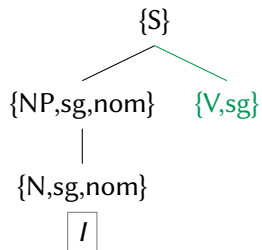


Finally, a precedence general constraint for verbs and nouns:

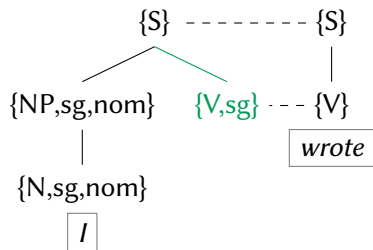




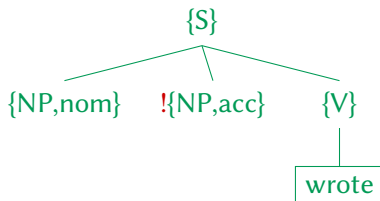
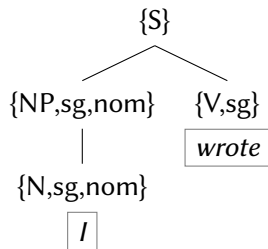
TUCO example



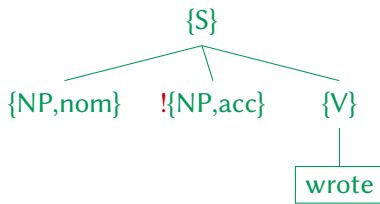
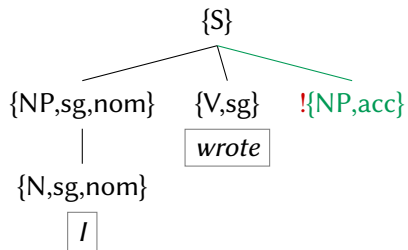
TUCO example



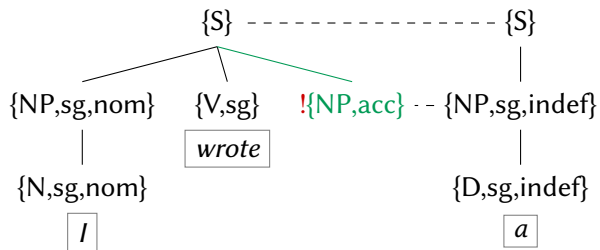
TUCO example



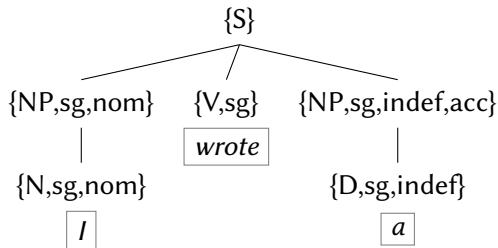
TUCO example



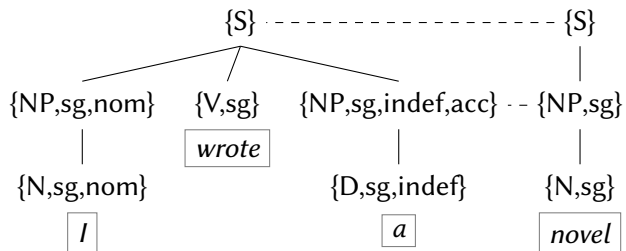
TUCO example



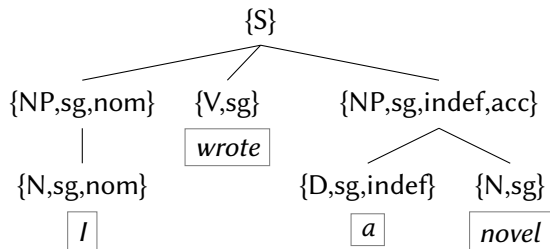
TUCO example



TUCO example

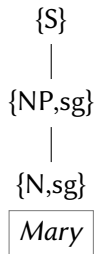


TUCO example

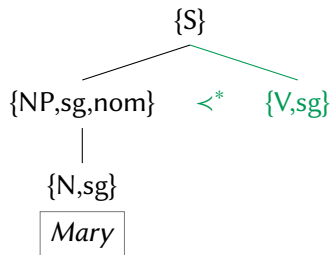


- 1 Introduction
- 2 TUCO in a nutshell
- 3 Gapping with TUCO**
- 4 Gapping resolution with TUCO
- 5 Summary

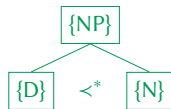
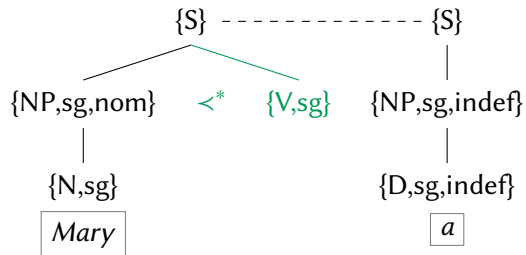
Gapping with TUCO



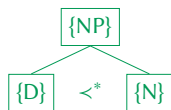
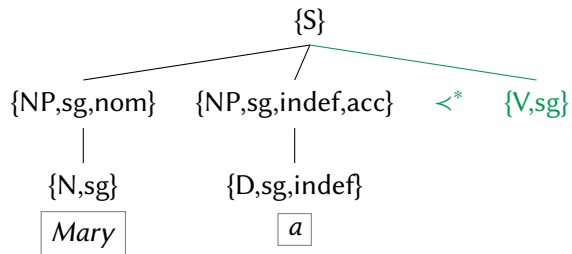
Gapping with TUCO



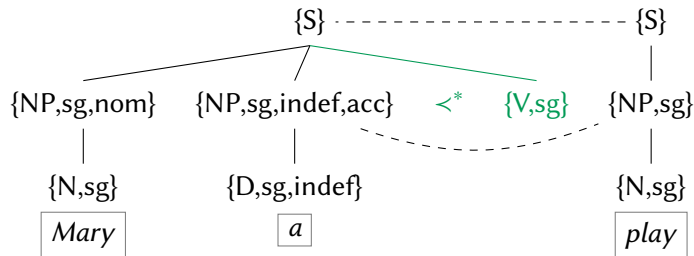
Gapping with TUCO



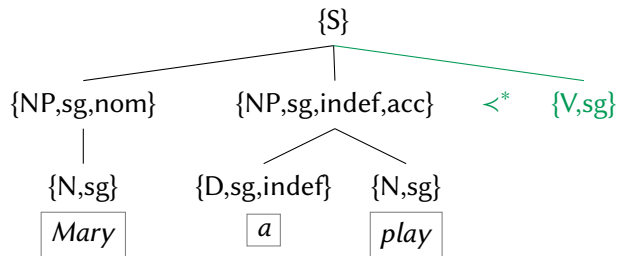
Gapping with TUCO



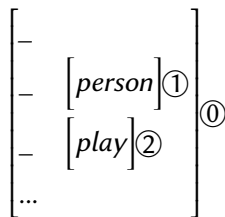
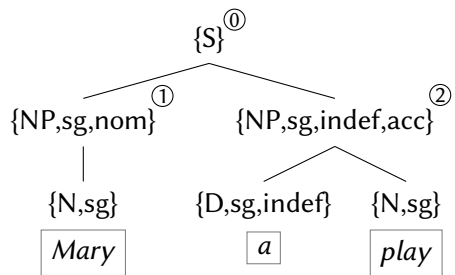
Gapping with TUCO



Gapping with TUCO



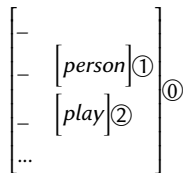
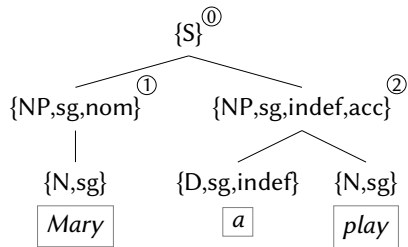
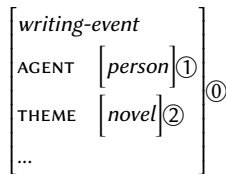
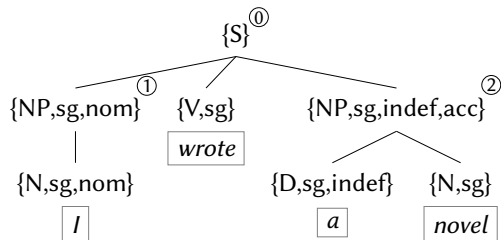
Gapping with TUCO



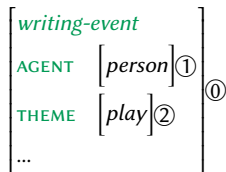
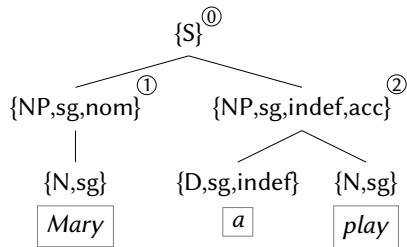
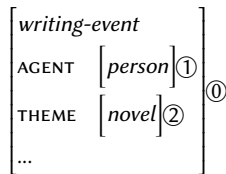
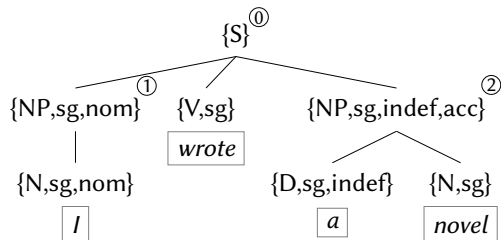
The semantics is composed in parallel!

- 1 Introduction
- 2 TUCO in a nutshell
- 3 Gapping with TUCO
- 4 Gapping resolution with TUCO**
- 5 Summary

Gapping resolution with TUCO



Gapping resolution with TUCO



Gapping resolution with TUCO: semantic approach

Naive semantic approach based on **unification with the overlap**:

$$R = (A \sqcap E) \sqcup E$$

$$\begin{bmatrix} \textit{writing-event} \\ \text{AGENT} \quad \begin{bmatrix} \textit{person} \end{bmatrix} \\ \text{THEME} \quad \begin{bmatrix} \textit{play} \end{bmatrix} \\ \dots \end{bmatrix} = \left(\begin{bmatrix} \textit{writing-event} \\ \text{AGENT} \quad \begin{bmatrix} \textit{person} \end{bmatrix} \\ \text{THEME} \quad \begin{bmatrix} \textit{novel} \end{bmatrix} \\ \dots \end{bmatrix} \sqcap \begin{bmatrix} - \\ - \quad \begin{bmatrix} \textit{person} \end{bmatrix} \\ - \quad \begin{bmatrix} \textit{play} \end{bmatrix} \\ \dots \end{bmatrix} \right) \sqcup \begin{bmatrix} - \\ - \quad \begin{bmatrix} \textit{person} \end{bmatrix} \\ - \quad \begin{bmatrix} \textit{play} \end{bmatrix} \\ \dots \end{bmatrix}$$

Gapping resolution with TUCO: semantic approach

Naive semantic approach based on **unification with the overlap**:

$$R = (A \sqcap E) \sqcup E$$

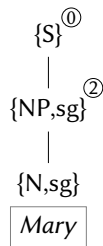
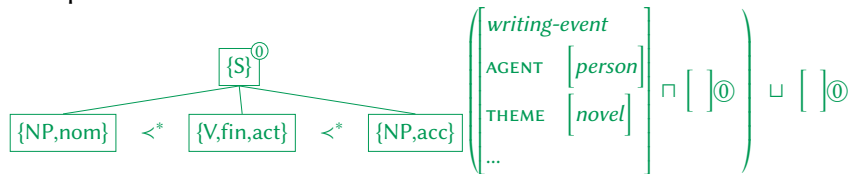
$$\begin{bmatrix} \textit{writing-event} \\ \text{AGENT} \quad \begin{bmatrix} \textit{person} \end{bmatrix} \\ \text{THEME} \quad \begin{bmatrix} \textit{play} \end{bmatrix} \\ \dots \end{bmatrix} = \left(\begin{bmatrix} \textit{writing-event} \\ \text{AGENT} \quad \begin{bmatrix} \textit{person} \end{bmatrix} \\ \text{THEME} \quad \begin{bmatrix} \textit{novel} \end{bmatrix} \\ \dots \end{bmatrix} \sqcap \begin{bmatrix} - \\ - \quad \begin{bmatrix} \textit{person} \end{bmatrix} \\ - \quad \begin{bmatrix} \textit{play} \end{bmatrix} \\ \dots \end{bmatrix} \right) \sqcup \begin{bmatrix} - \\ - \quad \begin{bmatrix} \textit{person} \end{bmatrix} \\ - \quad \begin{bmatrix} \textit{play} \end{bmatrix} \\ \dots \end{bmatrix}$$

But we need some syntax:

- violation of the major constituent constraint:
I wrote a long novel, Mary ~~wrote~~ a long play.
- illicit active-passive mismatch:
A novel was written by me, Mary ~~wrote~~ a play.

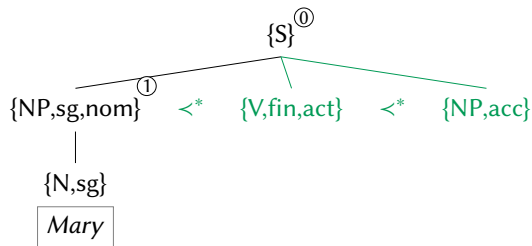
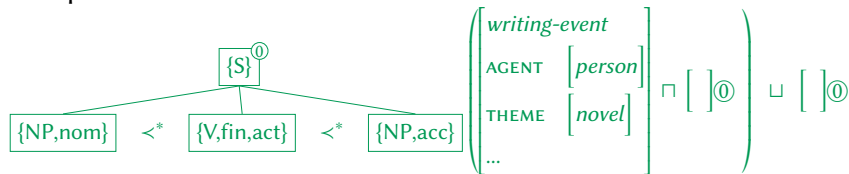
Gapping resolution with TUCO: context constraints

Syntactically grounded & contextually induced constraints with overlap unification:



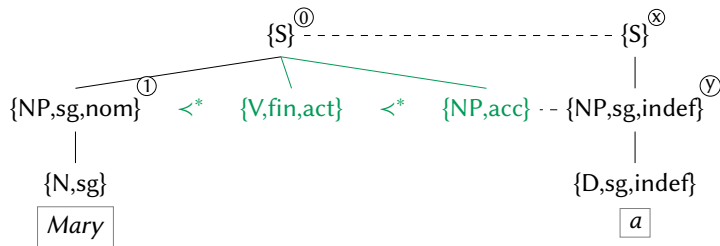
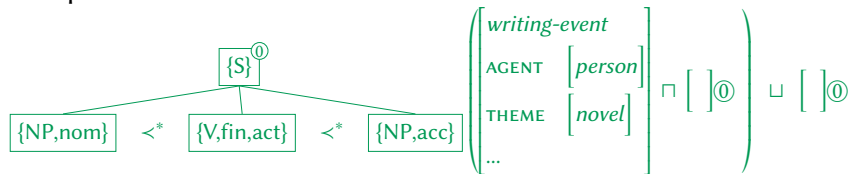
Gapping resolution with TUCO: context constraints

Syntactically grounded & contextually induced constraints with overlap unification:



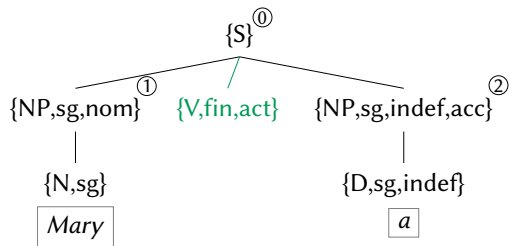
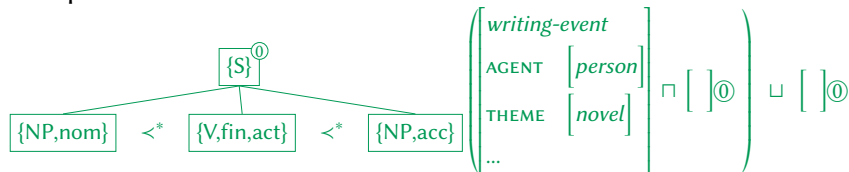
Gapping resolution with TUCO: context constraints

Syntactically grounded & contextually induced constraints with overlap unification:



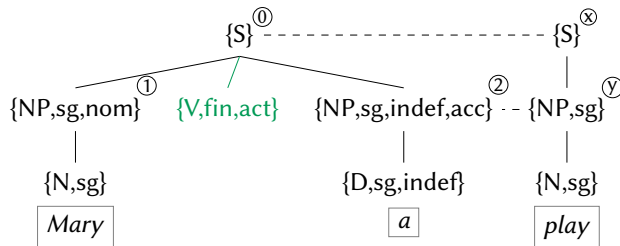
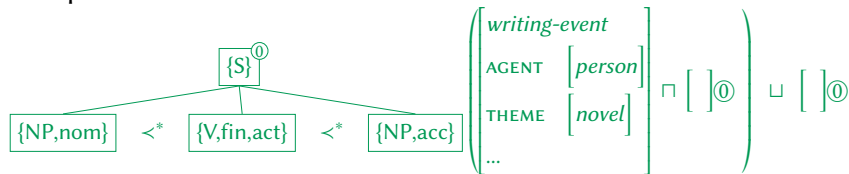
Gapping resolution with TUCO: context constraints

Syntactically grounded & contextually induced constraints with overlap unification:



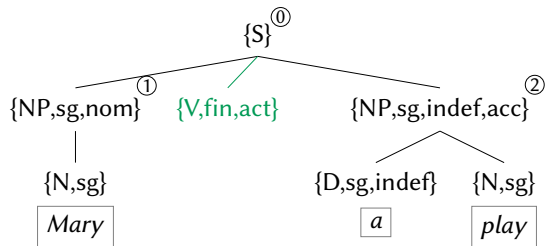
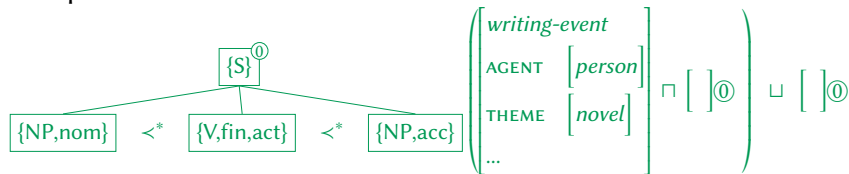
Gapping resolution with TUCO: context constraints

Syntactically grounded & contextually induced constraints with overlap unification:



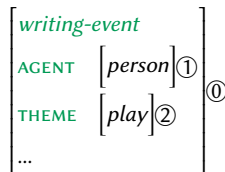
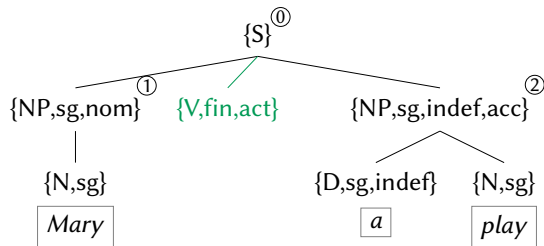
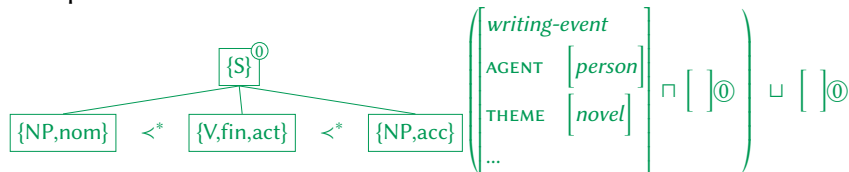
Gapping resolution with TUCO: context constraints

Syntactically grounded & contextually induced constraints with overlap unification:



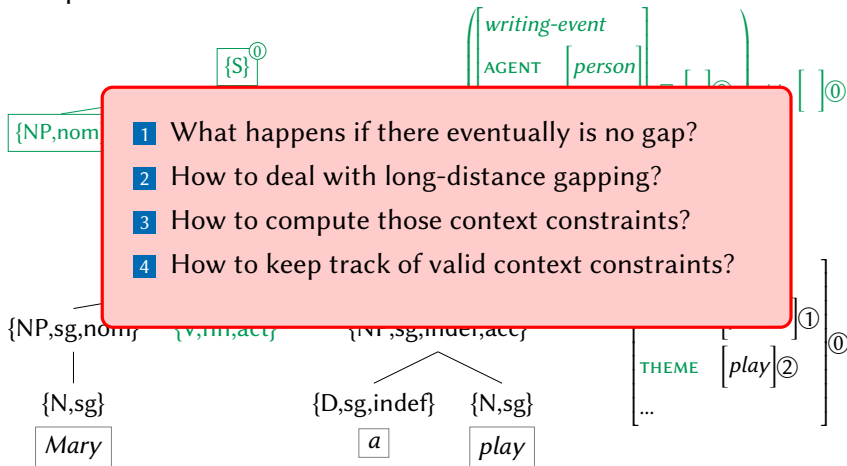
Gapping resolution with TUCO: context constraints

Syntactically grounded & contextually induced constraints with overlap unification:

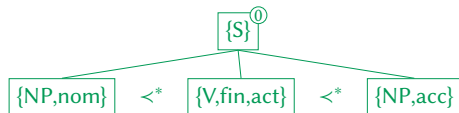


Gapping resolution with TUCO: context constraints

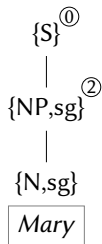
Syntactically grounded & contextually induced constraints with overlap unification:



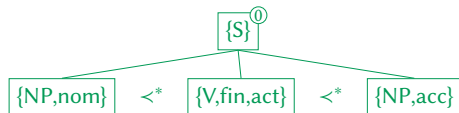
Canceling context constraints



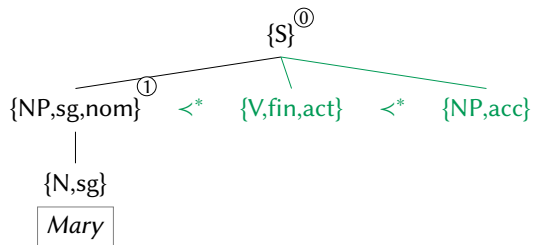
$$\left(\begin{array}{l} \textit{writing-event} \\ \text{AGENT} \quad [\textit{person}] \\ \text{THEME} \quad [\textit{novel}] \\ \dots \end{array} \right) \sqcap []^{\textcircled{1}} \sqcup []^{\textcircled{1}}$$



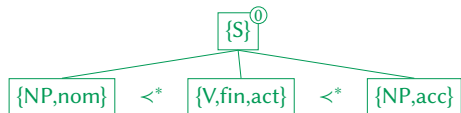
Canceling context constraints



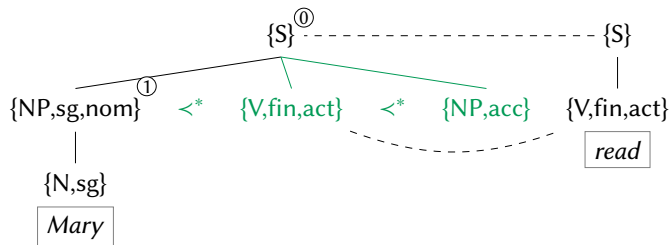
$$\left(\begin{array}{l} \textit{writing-event} \\ \text{AGENT} \quad [\textit{person}] \\ \text{THEME} \quad [\textit{novel}] \\ \dots \end{array} \right) \sqcap []^{\textcircled{0}} \sqcup []^{\textcircled{0}}$$



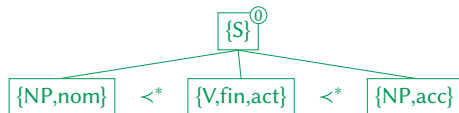
Canceling context constraints



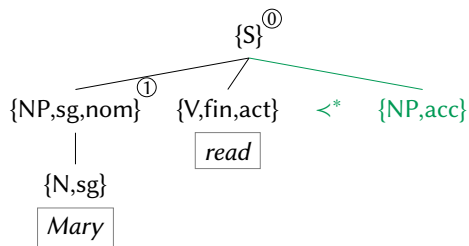
$$\left(\begin{array}{l} \textit{writing-event} \\ \text{AGENT} \quad [\textit{person}] \\ \text{THEME} \quad [\textit{novel}] \\ \dots \end{array} \right) \sqcap []^{\textcircled{0}} \sqcup []^{\textcircled{0}}$$



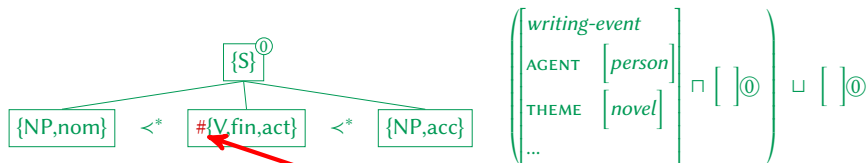
Canceling context constraints



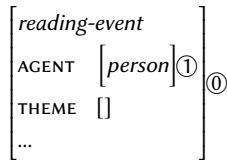
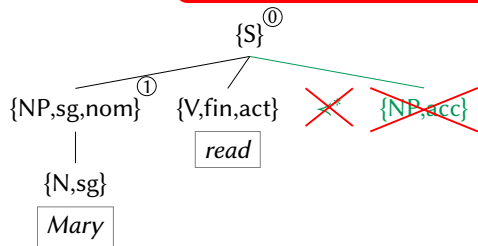
$$\left(\begin{array}{l} \textit{writing-event} \\ \text{AGENT} \quad \left[\textit{person} \right] \\ \text{THEME} \quad \left[\textit{novel} \right] \\ \dots \end{array} \right) \sqcap \left[\quad \right]^{\circledast} \sqcup \left[\quad \right]^{\circledast}$$



Canceling context constraints

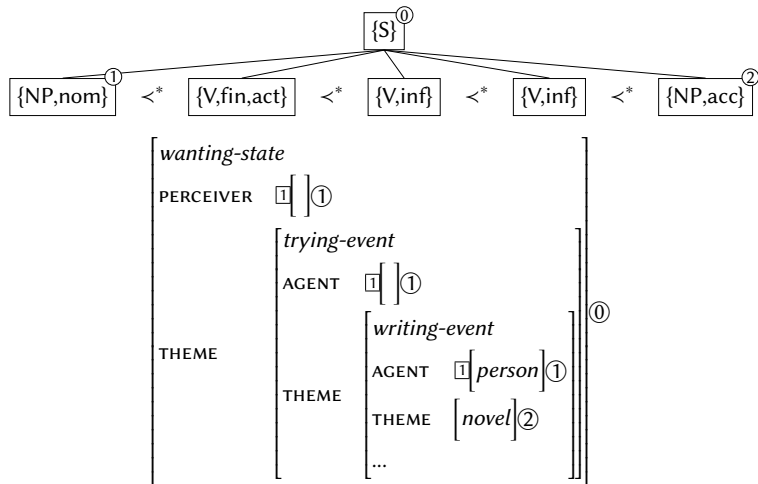


polarized predictions cannot be instantiated



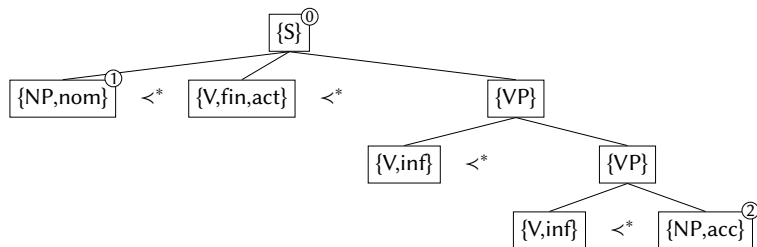
Resolution of long-distance gapping with TUCO

- (2) I want to try to write a novel.
Mary ~~wants to try to write~~ a play.



Resolution of long-distance gapping with TUCO

- (2) I want to try to write a novel.
Mary ~~wants to try to write~~ a play.



- 1 Introduction
- 2 TUCO in a nutshell
- 3 Gapping with TUCO
- 4 Gapping resolution with TUCO
- 5 Summary**

- “Gapping without surprise” implemented with TUCO:
 - this means: neither dedicated structure nor dedicated grammar rules
 - gapping resolution via local extension of the constraint set

- “Gapping without surprise” implemented with TUCO:
 - this means: neither dedicated structure nor dedicated grammar rules
 - gapping resolution via local extension of the constraint set
- Many open questions, of which I only mentioned a few:
 - How to compute those context constraints?
 - How to keep track of valid context constraints?
 - How to avoid the violation of the major constituent constraint?
 - How to deal with binding effects?
 - How to extend this to other ellipsis phenomena?
 - **[Your question goes here]**

