UPPSALA UNIVERSITET

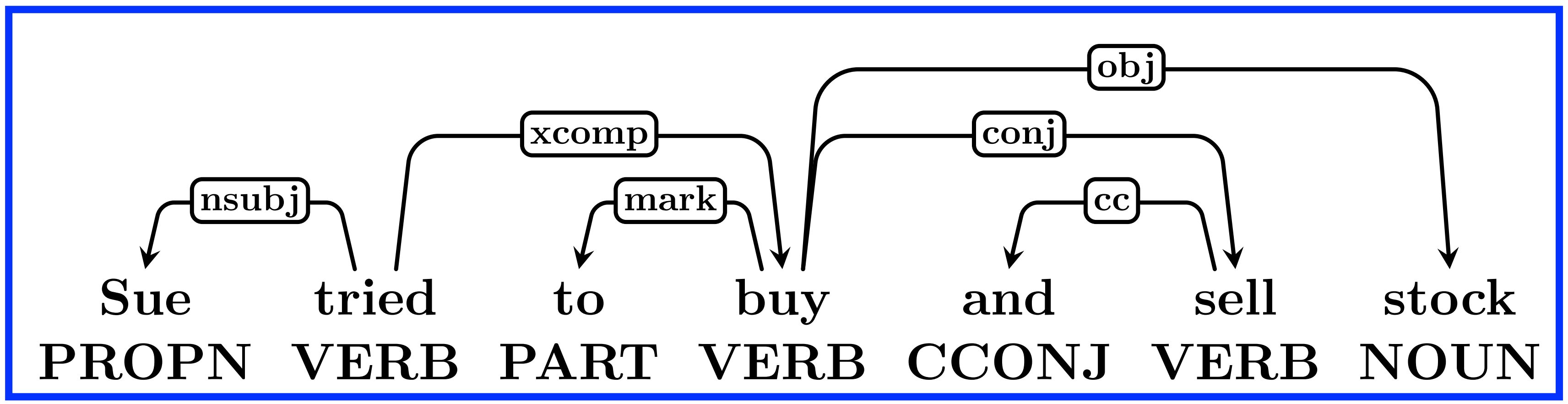
Enhanced Universal Dependencies: Enhancing Treebanks and Open Issues

Joakim Nivre and Sebastian Schuster

Uppsala University and Stanford University

Based on collaborative work with Marie Candito, Filip Ginter, Bruno Guillaume, Jenna Kanerva, Paola Marongiu, Simonetta Montemagni, Guy Perrier, Nathan Schneider, Djamé Seddah, and Maria Simi

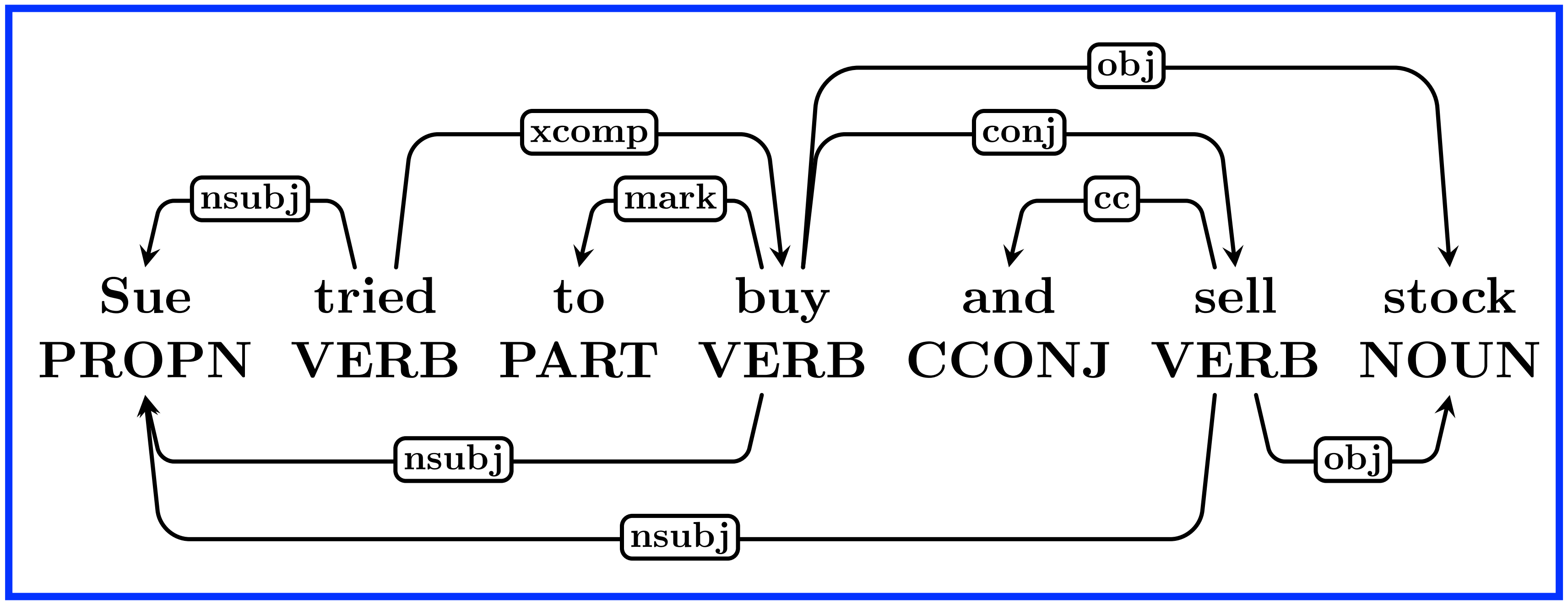
Basic Dependencies



Properties of the basic syntactic representation:

- Spanning tree over the words of the sentence
- One-to-one mapping from words to nodes – no empty nodes
- Every word related to (at most) one other word
- Underspecified representation of predicate-argument structure
- Suitable for parsing but not for (all) downstream applications

Enhanced Dependencies



Properties of the enhanced syntactic representation:

- General graph structure – not a tree (and not spanning)
- Partial mapping from words to nodes – and vice versa
- Not a monotonic extension of basic dependencies
- Disambiguates aspects of predicate-argument structure
- Collapses paths into single arcs – for practical convenience

Enhancements in UD v2

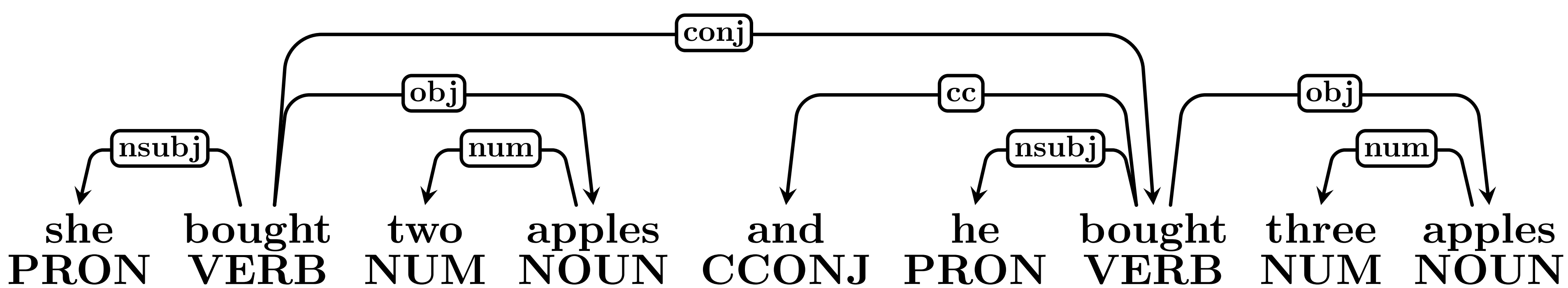
1. Null nodes for elided predicates
2. Shared heads and dependents in coordination
3. Added subject relations in control and raising
4. Coreference in relative clause constructions
5. Augmented modifier relations

Ellipsis

Ellipsis in basic dependencies:

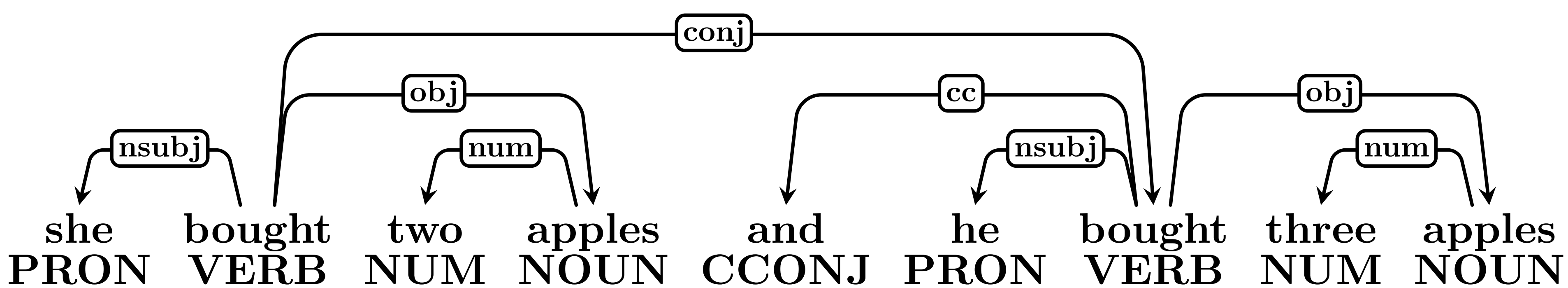
1. If the elided element has no overt dependents, we do nothing.
2. If the elided element has overt dependents, we promote one of these to take the role of the head.
3. If the elided element is a predicate and the promoted element a core argument or modifier, we use the orphan relation to attach other non-functional dependents to the promoted head.

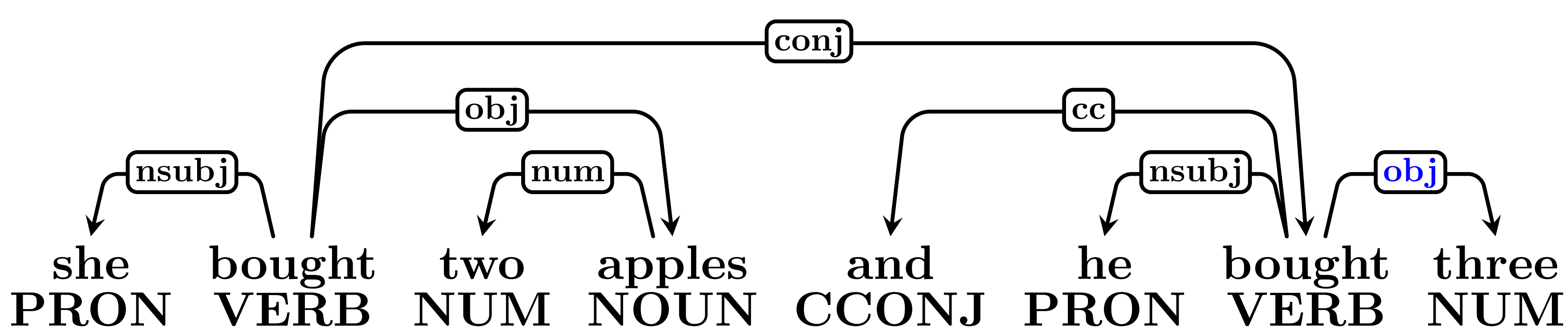
Ellipsis



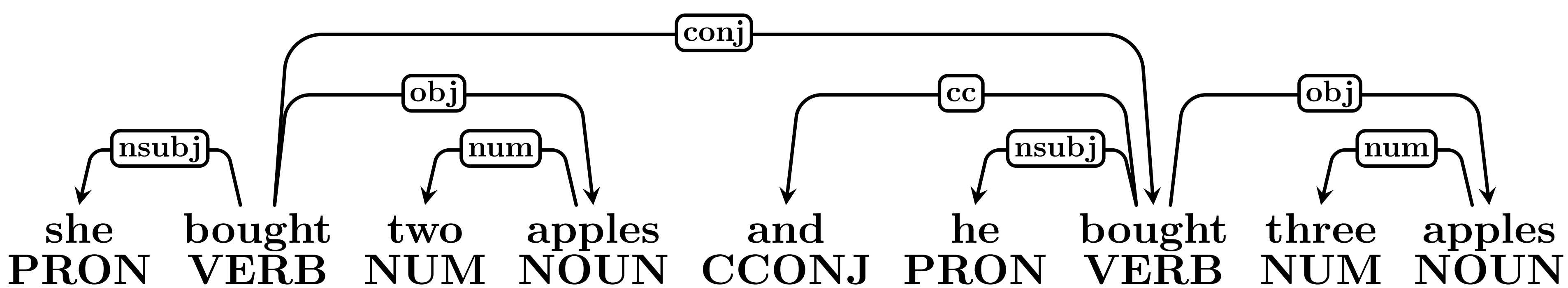
conj
obj
cc
obj
nsubj
num
nsubj
num
she bought two apples and he bought three apples
PRON VERB NUM NOUN CCONJ PRON VERB NUM NOUN

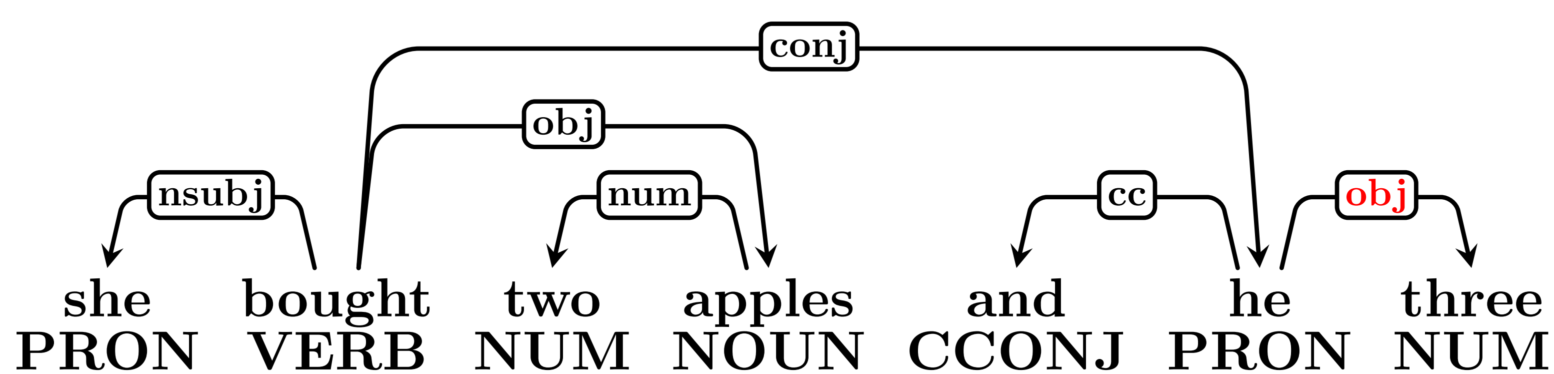
Ellipsis


conj
obj
cc
obj
nsubj
num
nsubj
num
she bought two apples and he bought three apples
PRON VERB NUM NOUN CCONJ PRON VERB NUM NOUN

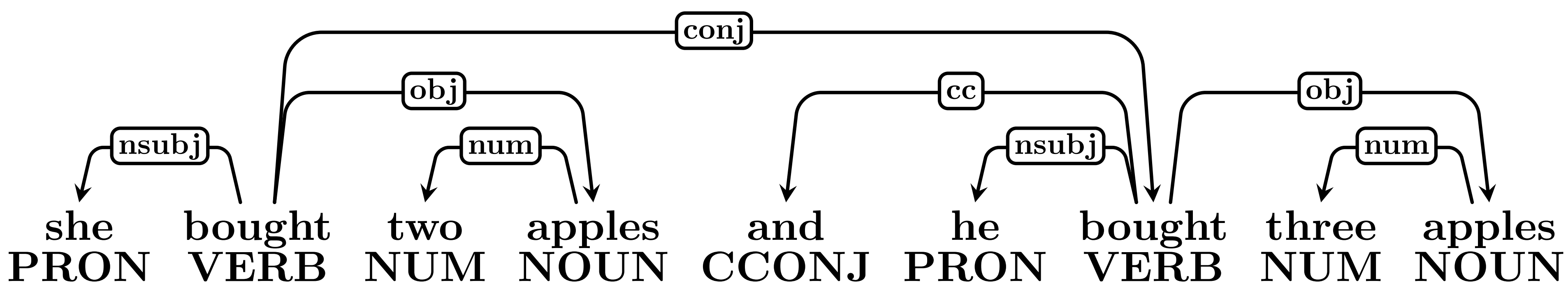

conj
obj
cc
nsubj
num
nsubj
obj
she bought two apples and he bought three
PRON VERB NUM NOUN CCONJ PRON VERB NUM

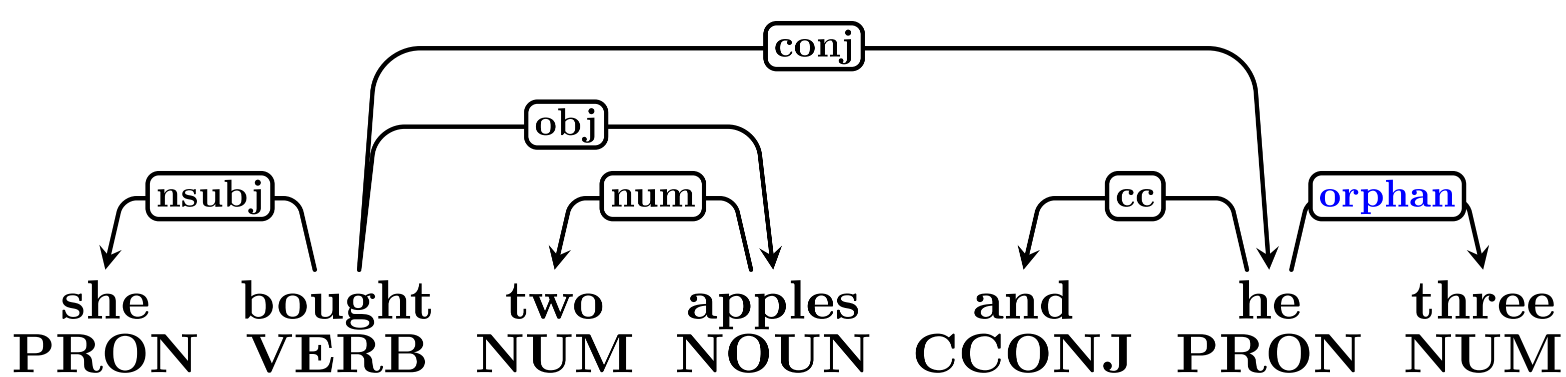
Ellipsis


conj
obj
cc
obj
nsubj
num
nsubj
num
she PRON
bought VERB
two NUM
apples NOUN
and CCONJ
he PRON
bought VERB
three NUM
apples NOUN


conj
obj
nsubj
num
cc
obj
she PRON
bought VERB
two NUM
apples NOUN
and CCONJ
he PRON
three NUM

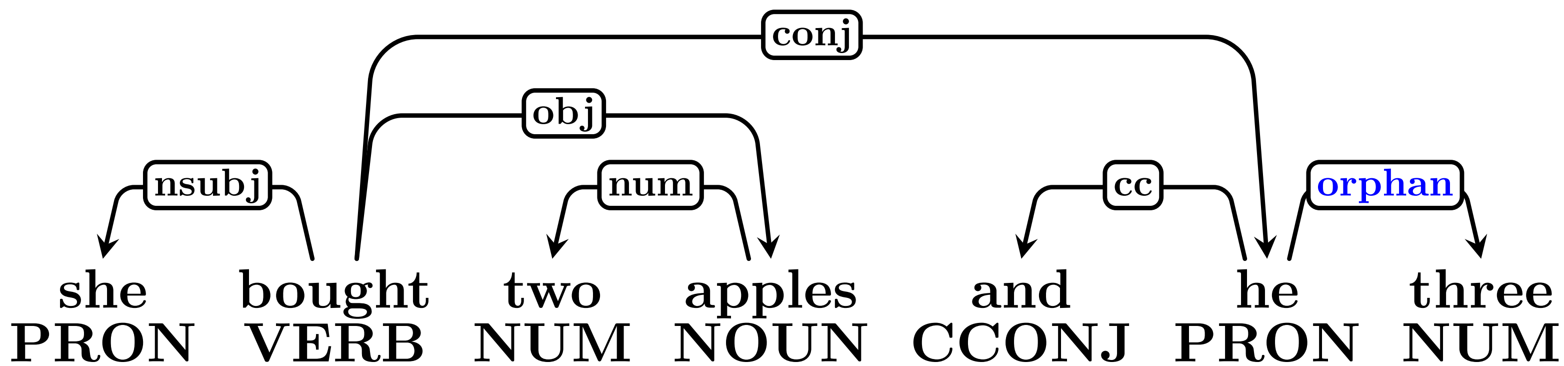
Ellipsis


conj
obj
cc
obj
nsubj
num
nsubj
num
she PRON
bought VERB
two NUM
apples NOUN
and CCONJ
he PRON
bought VERB
three NUM
apples NOUN

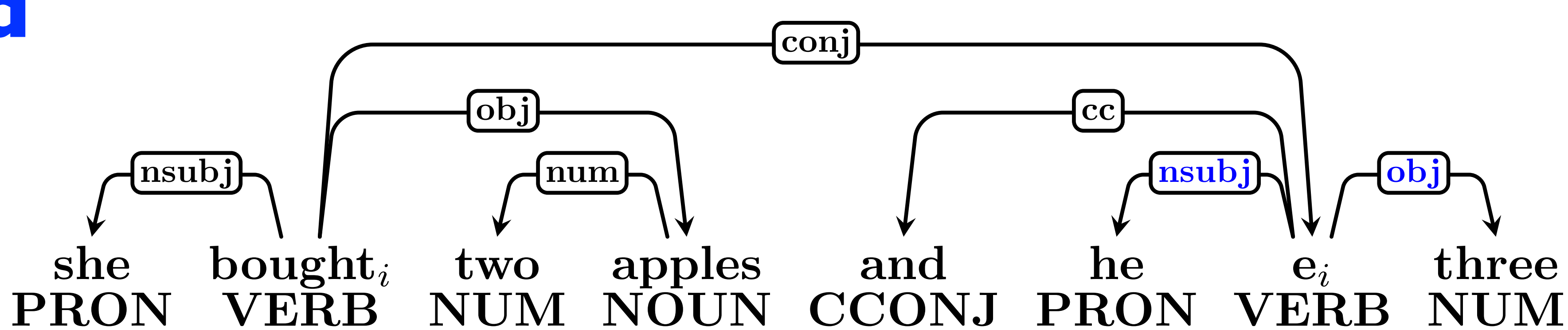

conj
obj
nsubj
num
cc
orphan
she PRON
bought VERB
two NUM
apples NOUN
and CCONJ
he PRON
three NUM

Ellipsis

Basic

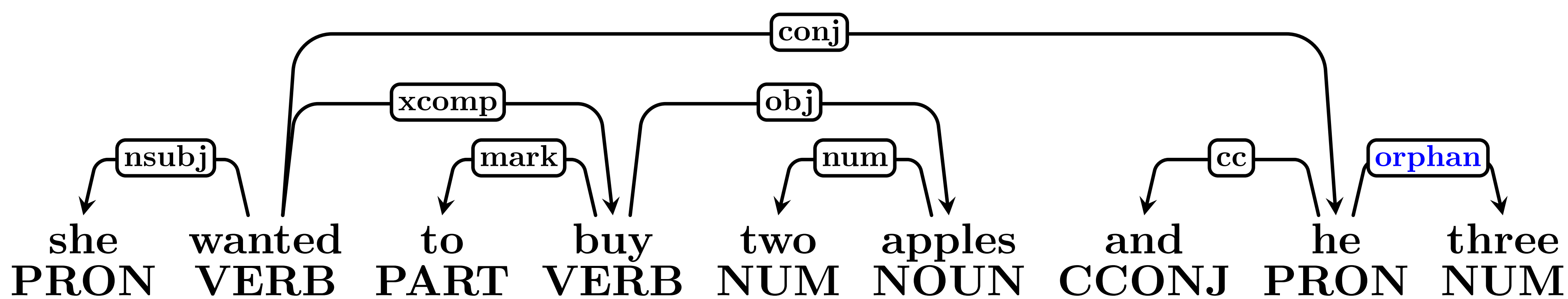


Enhanced

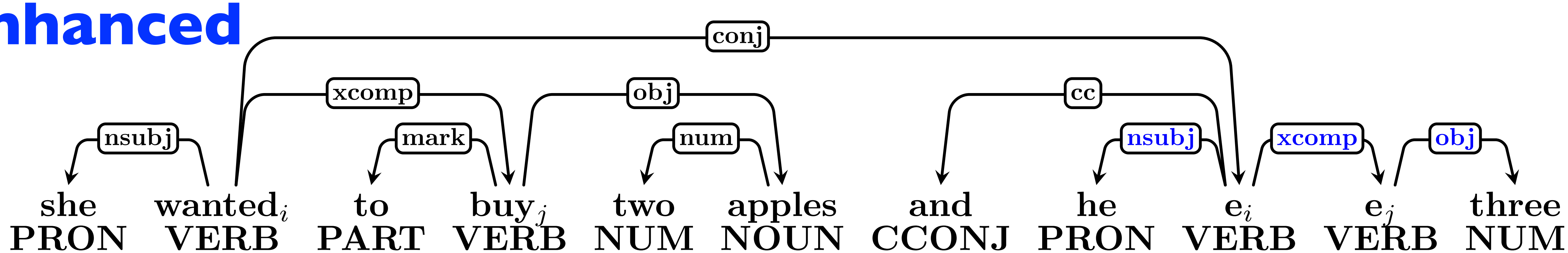


Ellipsis

Basic



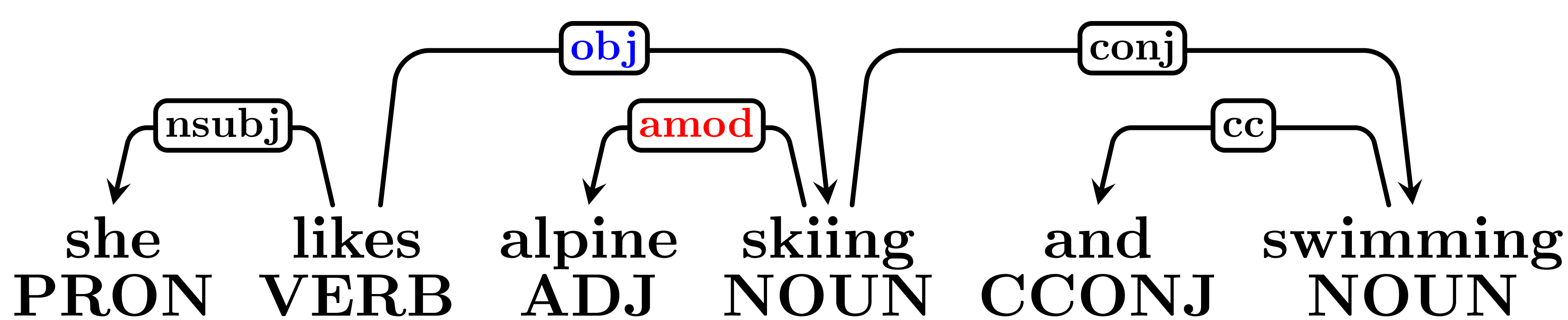
Enhanced



Coordination

Basic dependencies underspecify dependency relations into and out of coordinated phrases:

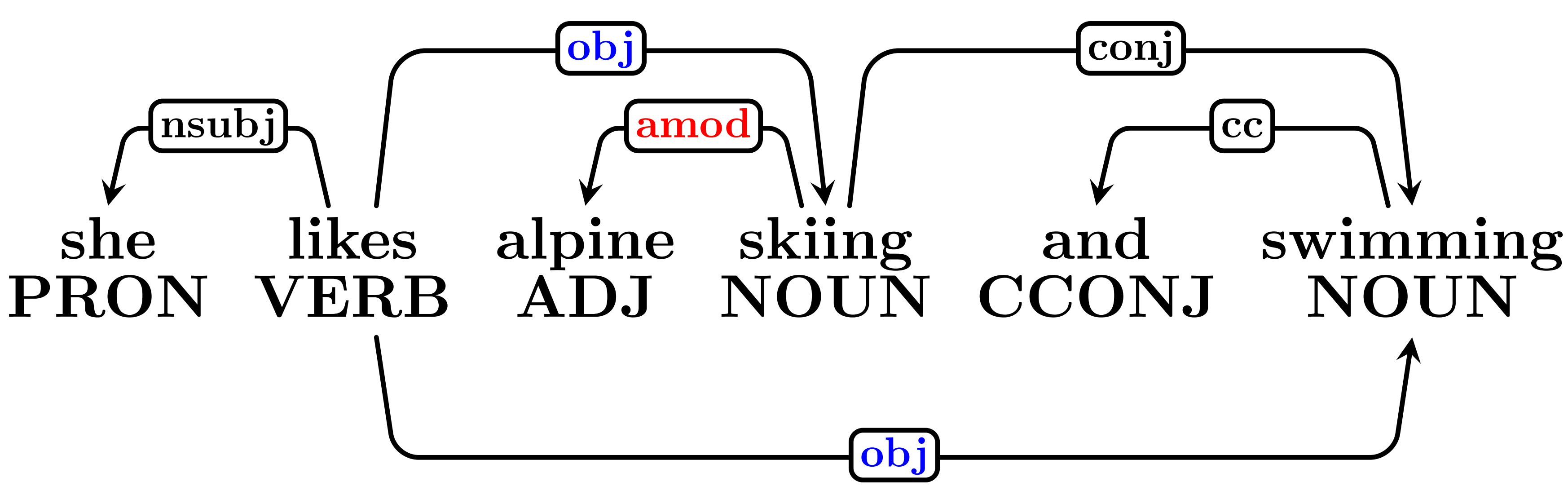
1. The shared head is attached only to the first conjunct (unambiguous).
2. Shared dependents are attached only to the first conjunct (ambiguous).



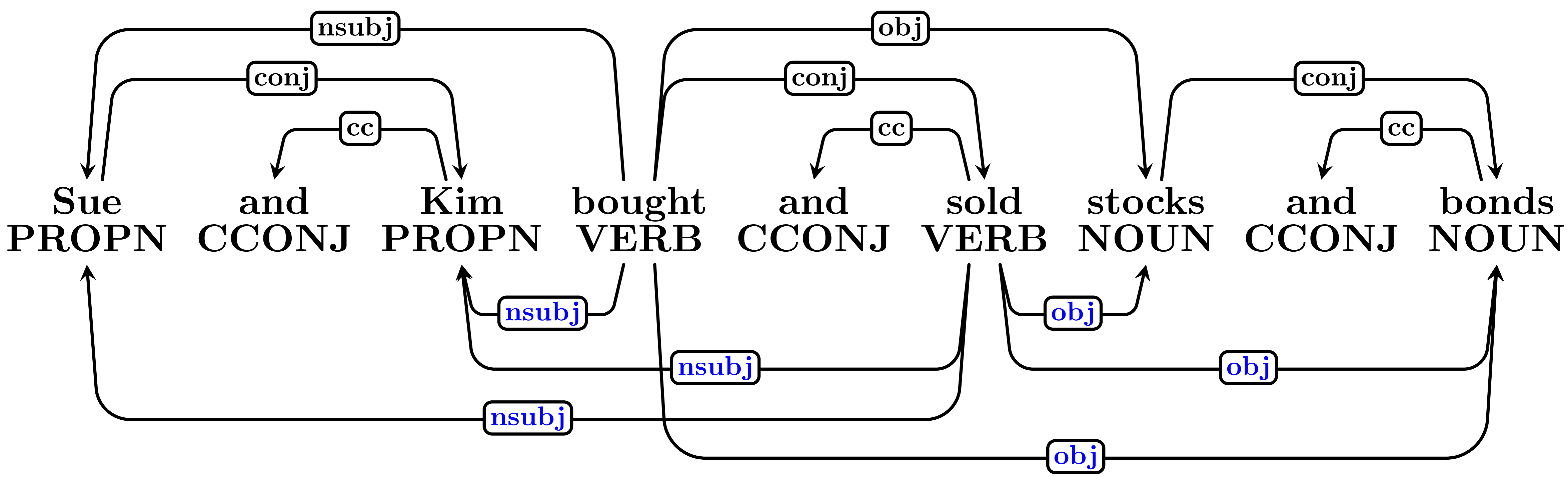
Coordination

Basic dependencies underspecify dependency relations into and out of coordinated phrases:

1. The shared head is attached only to the first conjunct (unambiguous).
2. Shared dependents are attached only to the first conjunct (ambiguous).



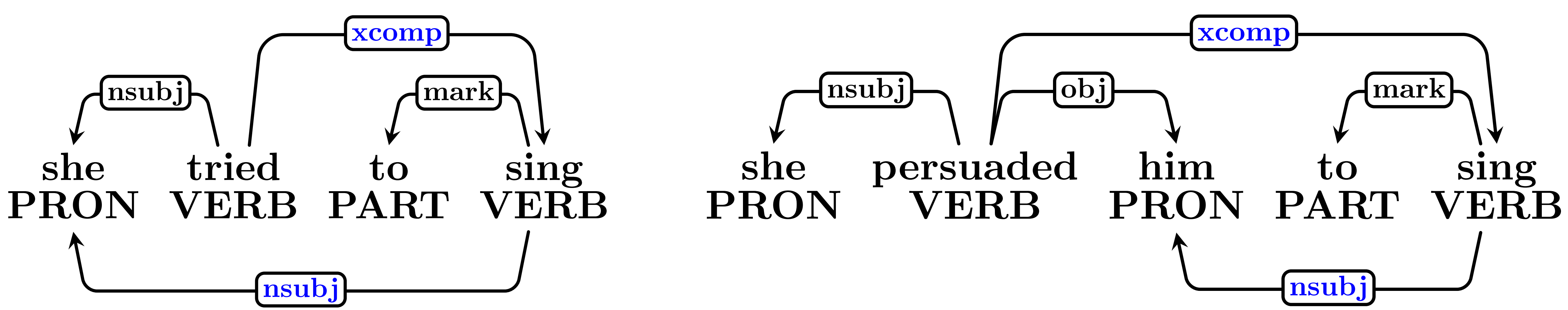
Coordination



Added Subjects

Enhanced dependencies add an explicit subject relation out of open clausal complements (xcomp)

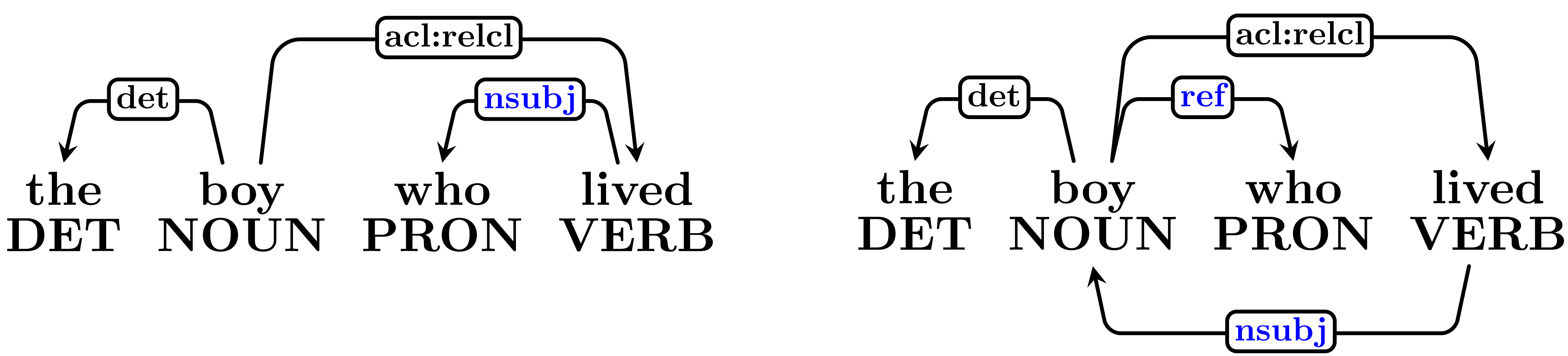
1. Subject relation to raised subject under raising verbs
2. Subject relation to controller under control verbs
3. Subject relation subject or object in nonverbal predication



Relative Clauses

Enhanced dependencies adds two relations:

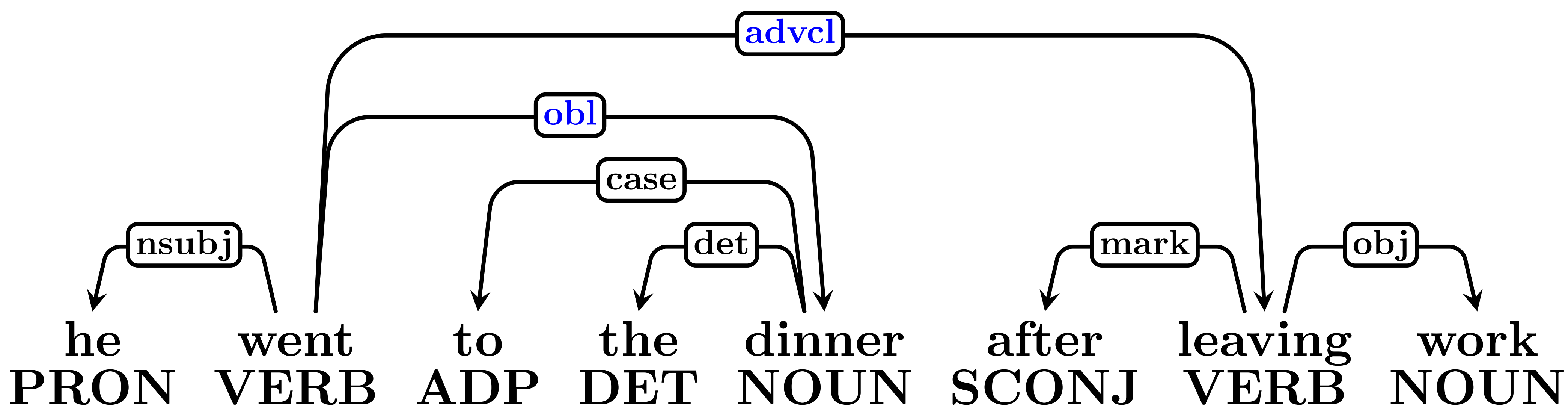
1. Co-reference relation from antecedent to relative pronoun
2. Core argument relation from relative clause predicate to antecedent



Augmented Labels

Enhanced dependencies augments relation labels:

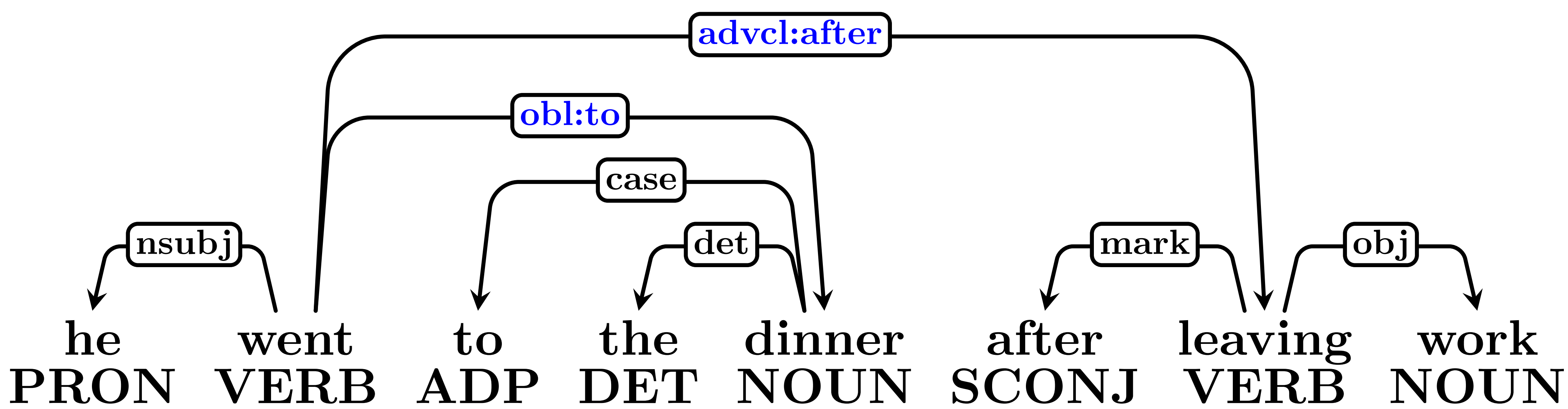
1. Adds case markers (adpositions) for obl and nmod
2. Adds markers (conjunctions) for advcl



Augmented Labels

Enhanced dependencies augments relation labels:

1. Adds case markers (adpositions) for obl and nmod
2. Adds markers (conjunctions) for advcl



Enhanced UD Treebanks

Enhanced UD Treebanks

UD v2.2: 5 out of 102 treebanks

- English (EWT)
- Finnish (TDT, PUD)
- Latvian (LVTB)
- Russian (SynTagRus)

Enhanced UD Treebanks

UD v2.2: 5 out of 102 treebanks

- English (EWT)
- Finnish (TDT, PUD)
- Latvian (LVTB)
- Russian (SynTagRus)

Case study on bootstrapping enhanced dependencies

- Joint work with Filip Ginter, Jenna Kanerva, Paola Marongiu, Simonetta Montemagni, Maria Simi

Case Study

Case Study

Two enhancers:

- Stanford – rule-based system developed for English
- Turku – data-driven system trained on Finnish

Case Study

Two enhancers:

- Stanford – rule-based system developed for English
- Turku – data-driven system trained on Finnish

Two target languages:

- Swedish
- Italian

Case Study

Two enhancers:

- Stanford – rule-based system developed for English
- Turku – data-driven system trained on Finnish

Two target languages:

- Swedish
- Italian

Three enhanced dependency types:

- Added subject relations in raising and control constructions
- Shared heads and dependents in coordination
- Null nodes for elided predicates

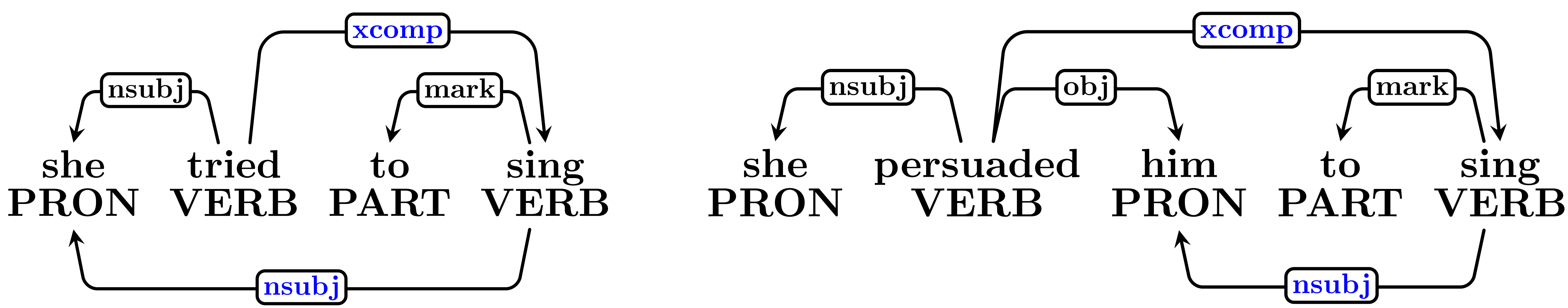
The Stanford System

- Based on English system by Schuster and Manning (2016)
- Pattern matching to detect structures to enhance
- Heuristics to predict enhanced dependencies
- Novel method for ellipsis by Schuster et al. (2018)

Stanford: Subjects

From any node attached as an xcomp to a higher predicate, add an nsubj dependency to:

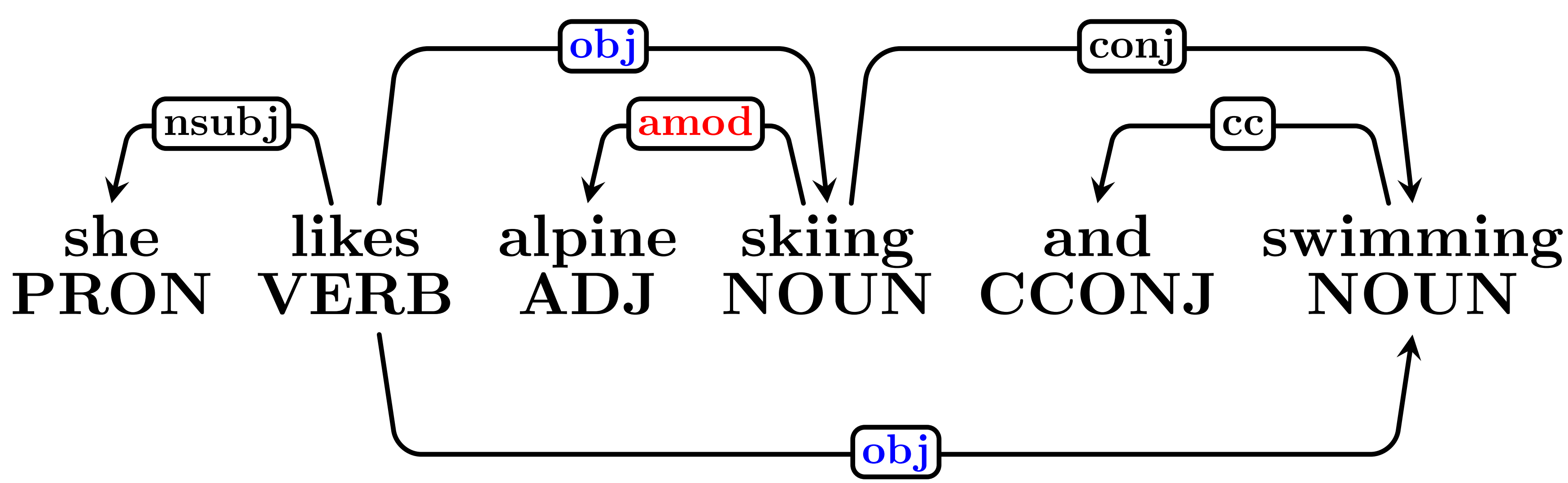
1. The obj of the predicate if such a dependent exists
2. The nsubj of the predicate otherwise (raising, subject control)



Stanford: Coordination

Two types of added dependencies:

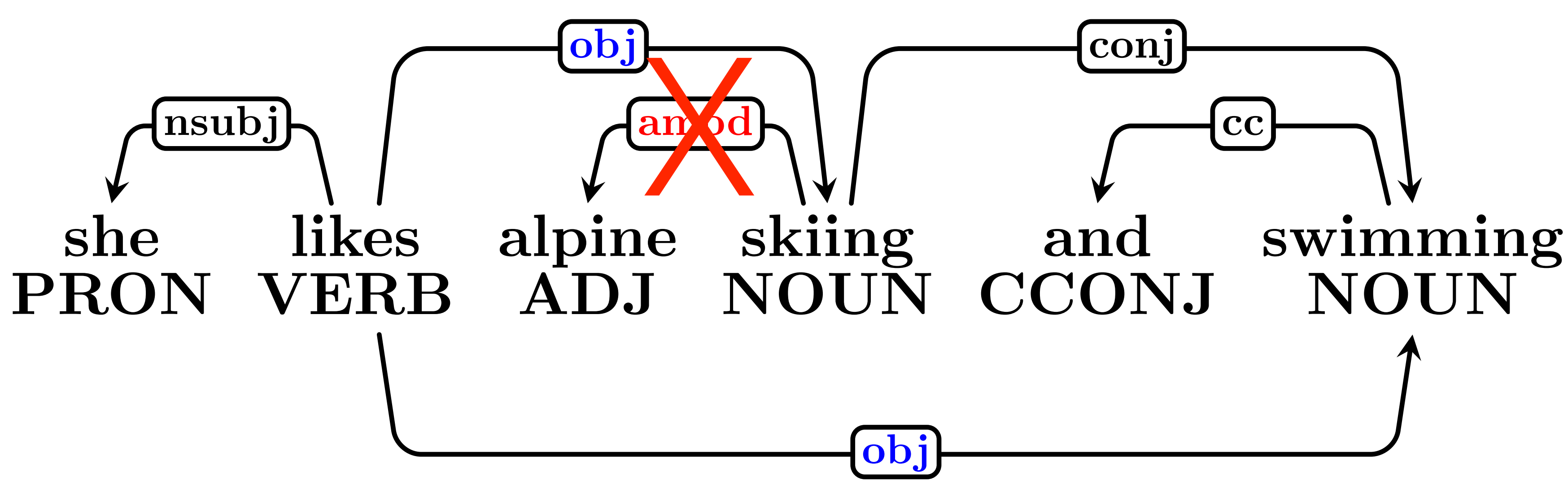
1. Shared heads for all coordinated elements
2. Shared dependents limited to core arguments of conjoined predicates: (i)obj, n/csubj, c/xcomp
3. Aims for high precision (rather than recall)



Stanford: Coordination

Two types of added dependencies:

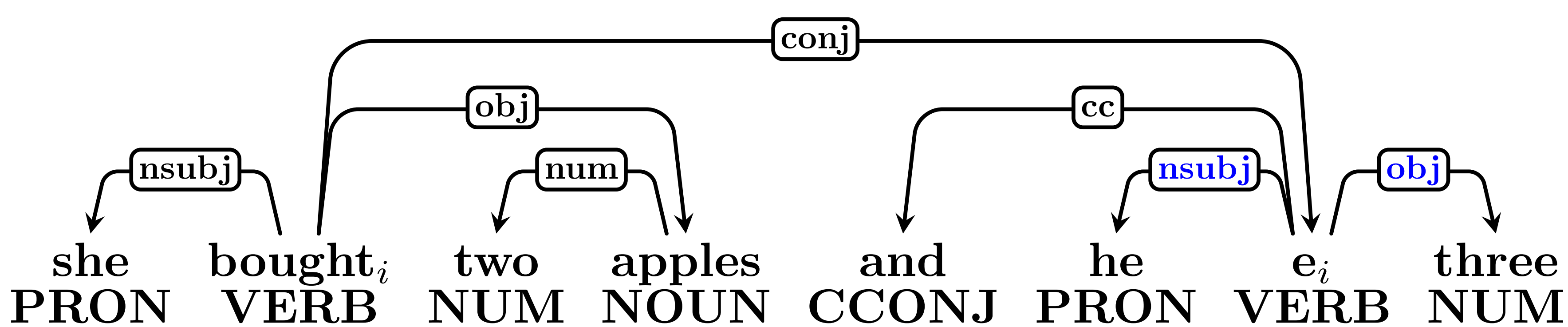
1. Shared heads for all coordinated elements
2. Shared dependents limited to core arguments of conjoined predicates: (i)obj, n/csubj, c/xcomp
3. Aims for high precision (rather than recall)



Stanford: Ellipsis

Enhanced dependencies augments relation labels:

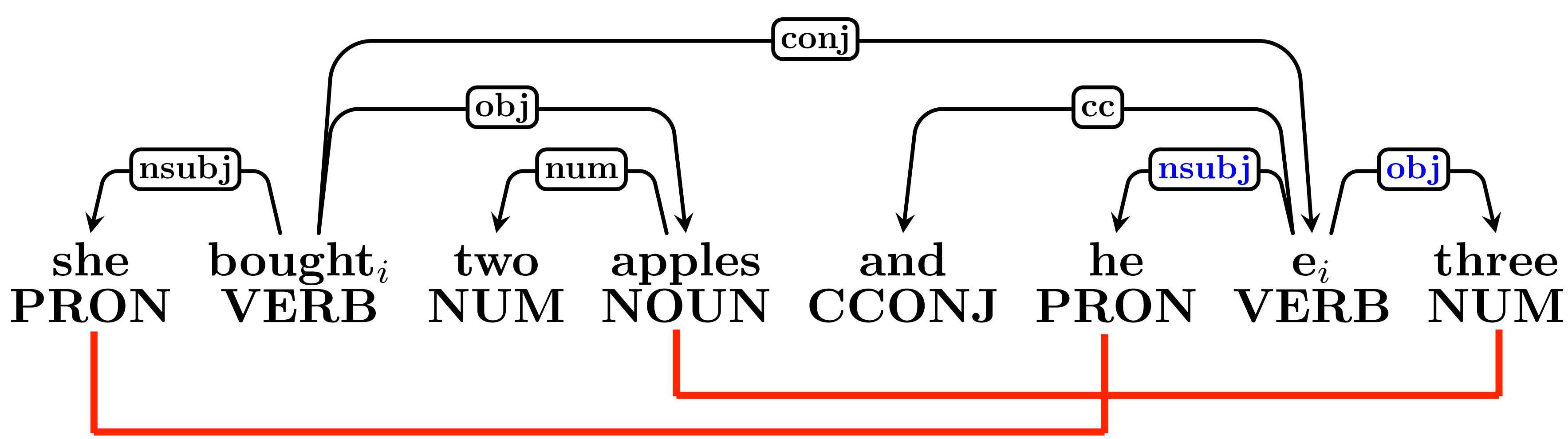
1. Align arguments and modifiers in the complete and gapped clause using similarity of word embeddings
2. Add null predicates corresponding to non-matched items and add dependencies based on matchings



Stanford: Ellipsis

Enhanced dependencies augments relation labels:

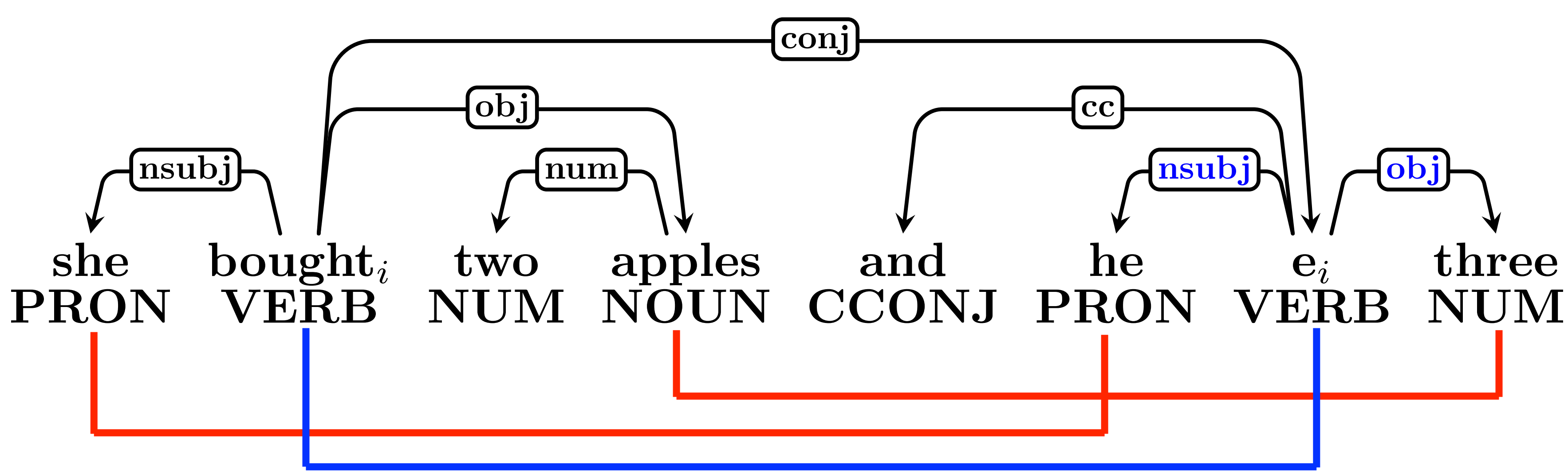
1. Align arguments and modifiers in the complete and gapped clause using similarity of word embeddings
2. Add null predicates corresponding to non-matched items and add dependencies based on matchings



Stanford: Ellipsis

Enhanced dependencies augments relation labels:

1. Align arguments and modifiers in the complete and gapped clause using similarity of word embeddings
2. Add null predicates corresponding to non-matched items and add dependencies based on matchings



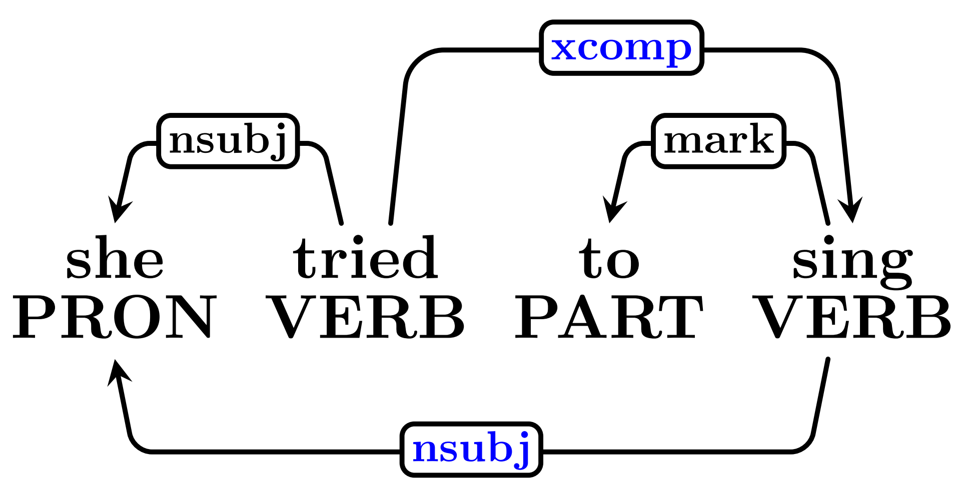
The Turku System

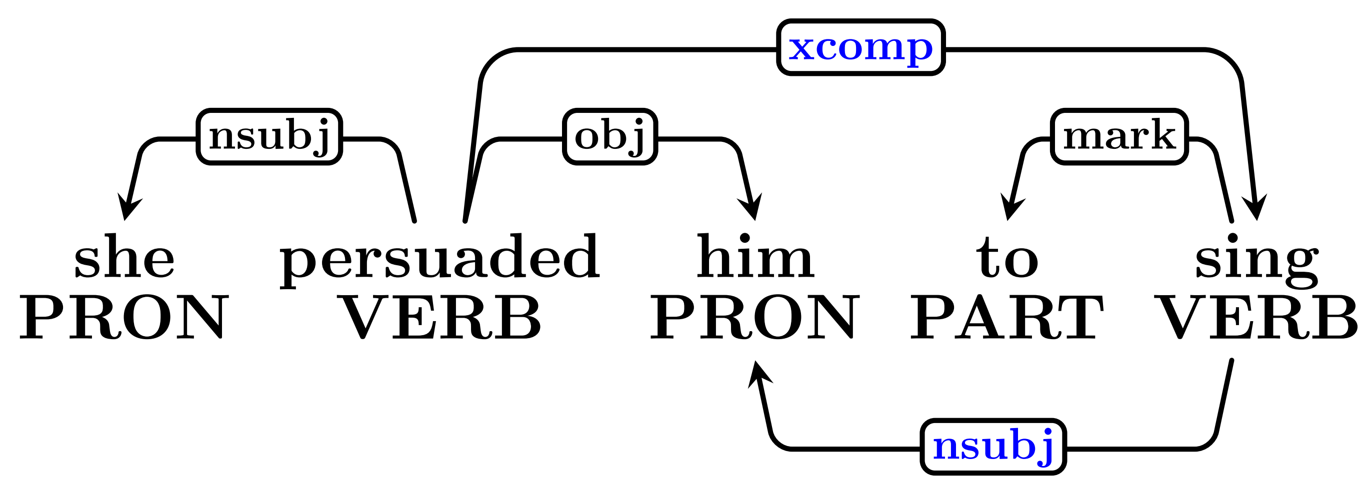
- Based on Finnish system by Nyblom et al. (2016)
- Pattern matching to detect structures to enhance
- SVM classifier selects candidate dependencies
- Language-specific features omitted for generality
- Does not handle null nodes for elided predicates

Turku: Subjects

From any infinitive verb attached as an xcomp to a higher predicate, consider adding an nsubj dependency to the nsubj of the predicate

1. Binary SVM classifier decides if dependency is added or not
2. Object control is not considered at all

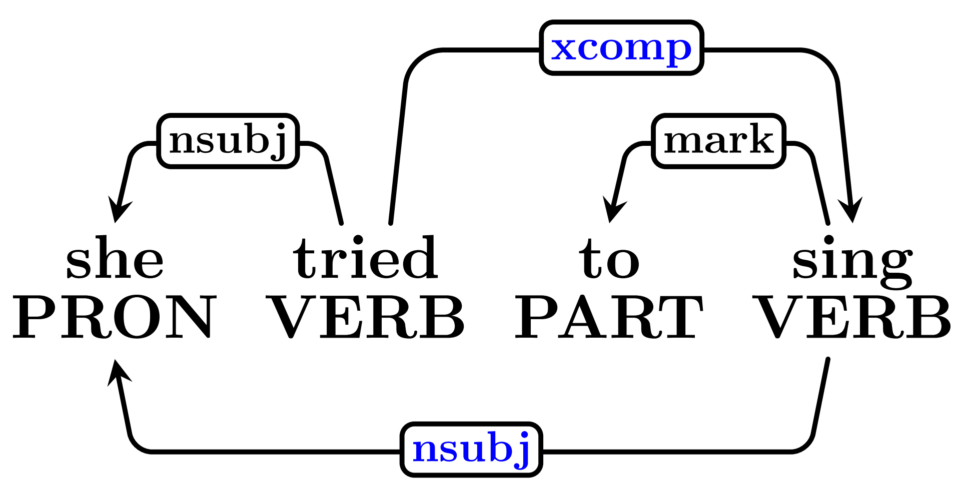


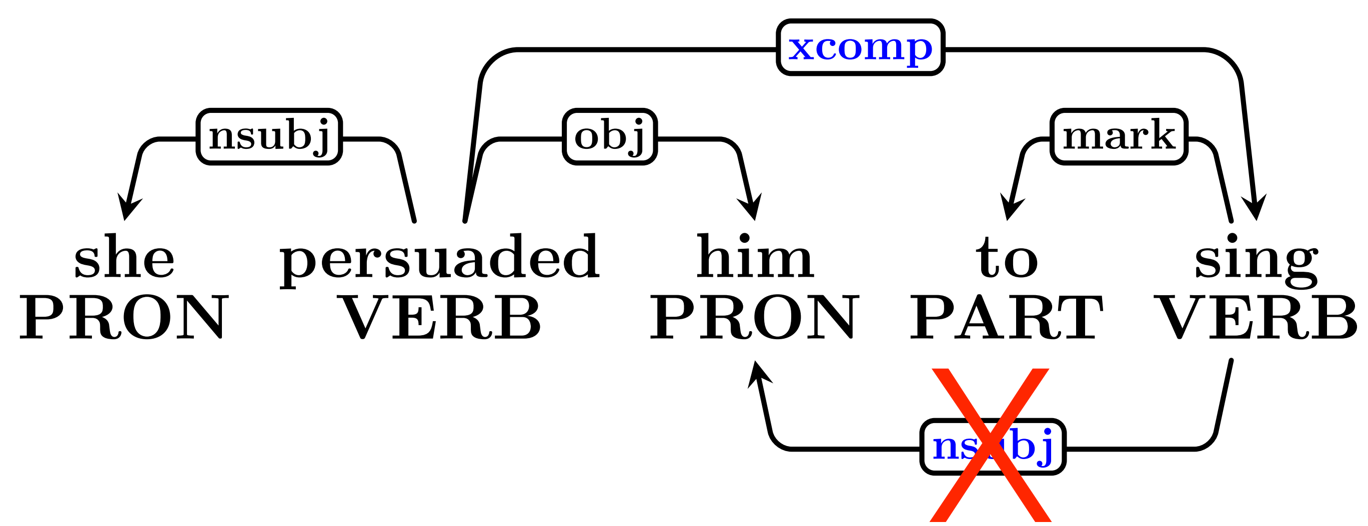


Turku: Subjects

From any infinitive verb attached as an xcomp to a higher predicate, consider adding an nsubj dependency to the nsubj of the predicate

1. Binary SVM classifier decides if dependency is added or not
2. Object control is not considered at all





Turku: Coordination

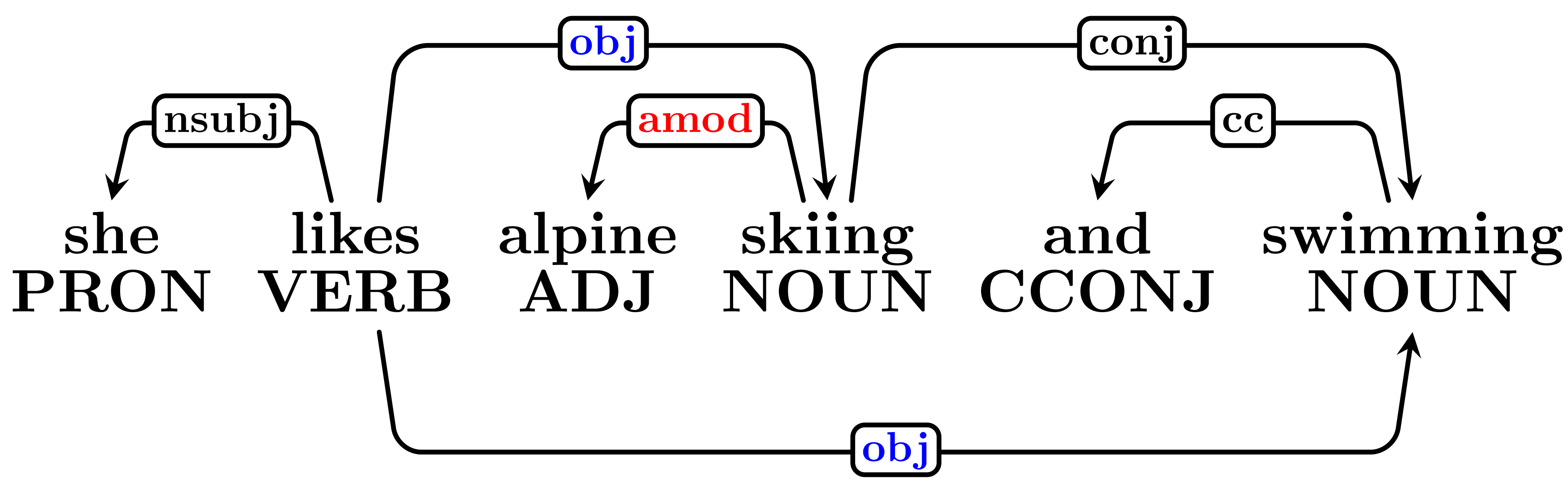
The head and all dependents of the first conjunct are considered candidate head/dependents of all conjuncts

1. SVM classifier selects dependency label or null
2. Aims for high recall (rather than precision)

Turku: Coordination

The head and all dependents of the first conjunct are considered candidate head/dependents of all conjuncts

1. SVM classifier selects dependency label or null
2. Aims for high recall (rather than precision)



Evaluation

Evaluation

Manual evaluation:

- 1000 sentences from the training set for subjects and coordination
- The entire training sets for ellipsis (rare)

Evaluation

Manual evaluation:

- 1000 sentences from the training set for subjects and coordination
- The entire training sets for ellipsis (rare)

Error classification:

- Basic errors – errors caused by incorrect basic dependencies
- Enhanced errors – errors in spite of correct basic dependencies

Evaluation

Manual evaluation:

- 1000 sentences from the training set for subjects and coordination
- The entire training sets for ellipsis (rare)

Error classification:

- Basic errors – errors caused by incorrect basic dependencies
- Enhanced errors – errors in spite of correct basic dependencies

Evaluation metrics:

- Precision – percentage of predicted dependencies that are valid
- Recall – percentage of valid dependencies predicted (relative)

Results

Results

	Subjects	
	Swe	
	S	T
Count	127	36
Precision	0.87	0.83
Recall (relative)	0.98	0.27
Basic errors	12	1
Enhanced errors	4	5

Results

	Subjects			
	Swe		Ita	
	S	T	S	T
Count	127	36	115	43
Precision	0.87	0.83	0.80	0.95
Recall (relative)	0.98	0.27	0.91	0.41
Basic errors	12	1	14	0
Enhanced errors	4	5	9	2

Results

	Subjects				Coordination	
	Swe		Ita		Swe	
	S	T	S	T	S	T
Count	127	36	115	43	559	981
Precision	0.87	0.83	0.80	0.95	0.94	0.91
Recall (relative)	0.98	0.27	0.91	0.41	0.55	0.97
Basic errors	12	1	14	0	25	28
Enhanced errors	4	5	9	2	9	69

Results

	Subjects				Coordination			
	Swe		Ita		Swe		Ita	
	S	T	S	T	S	T	S	T
Count	127	36	115	43	559	981	421	653
Precision	0.87	0.83	0.80	0.95	0.94	0.91	0.89	0.82
Recall (relative)	0.98	0.27	0.91	0.41	0.55	0.97	0.67	0.96
Basic errors	12	1	14	0	25	28	12	32
Enhanced errors	4	5	9	2	9	69	34	86

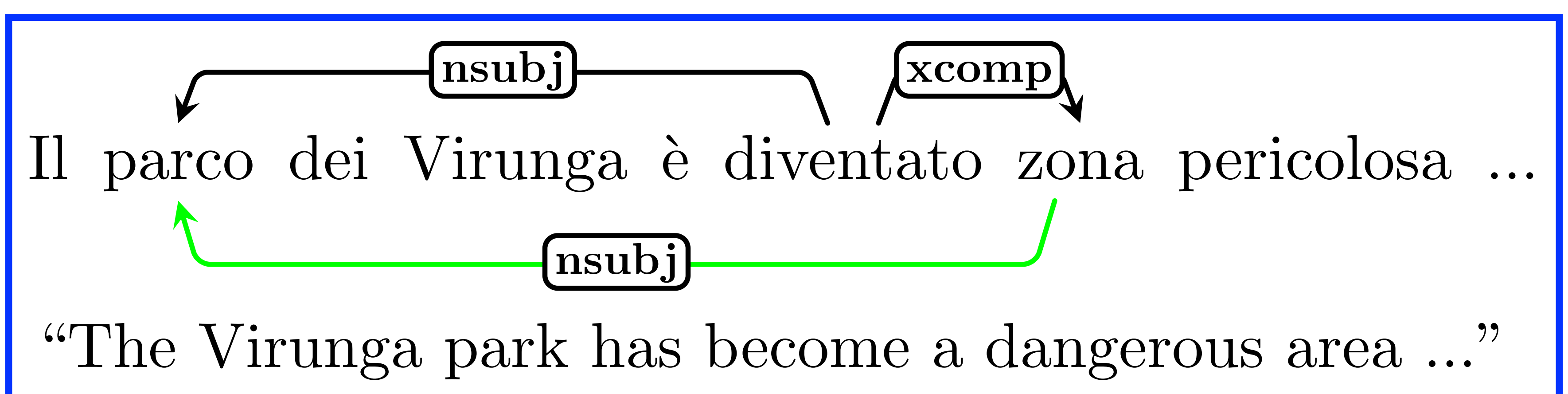
Results

	Subjects				Coordination				Ellipsis	
	Swe		Ita		Swe		Ita		Swe	Ita
	S	T	S	T	S	T	S	T	S	S
Count	127	36	115	43	559	981	421	653	112	162
Precision	0.87	0.83	0.80	0.95	0.94	0.91	0.89	0.82	0.85	0.76
Recall (relative)	0.98	0.27	0.91	0.41	0.55	0.97	0.67	0.96		
Basic errors	12	1	14	0	25	28	12	32	15	0
Enhanced errors	4	5	9	2	9	69	34	86	2	35

Error Analysis: Subjects

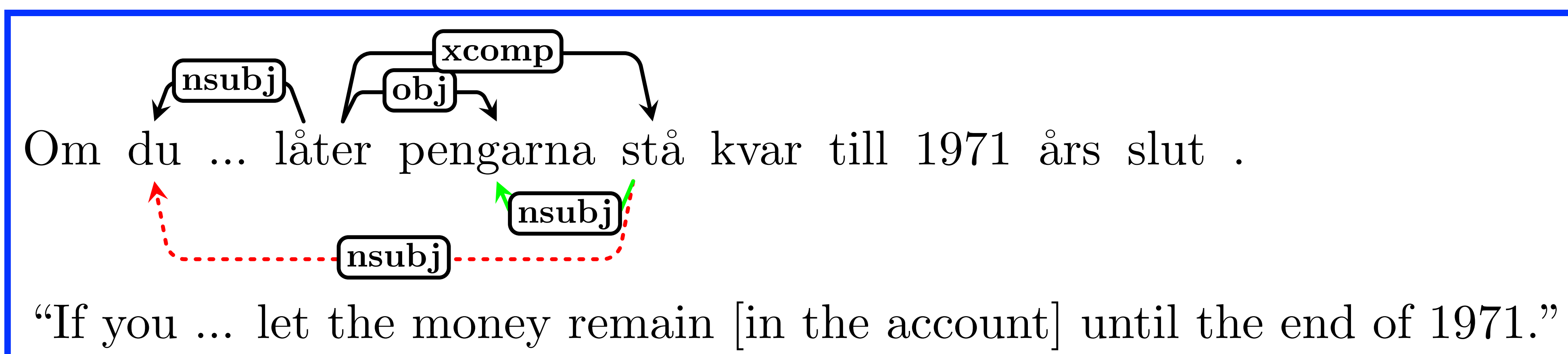
Error Analysis: Subjects

- Stanford has higher recall for both languages because it considers all xcomp nodes (not just infinitives)



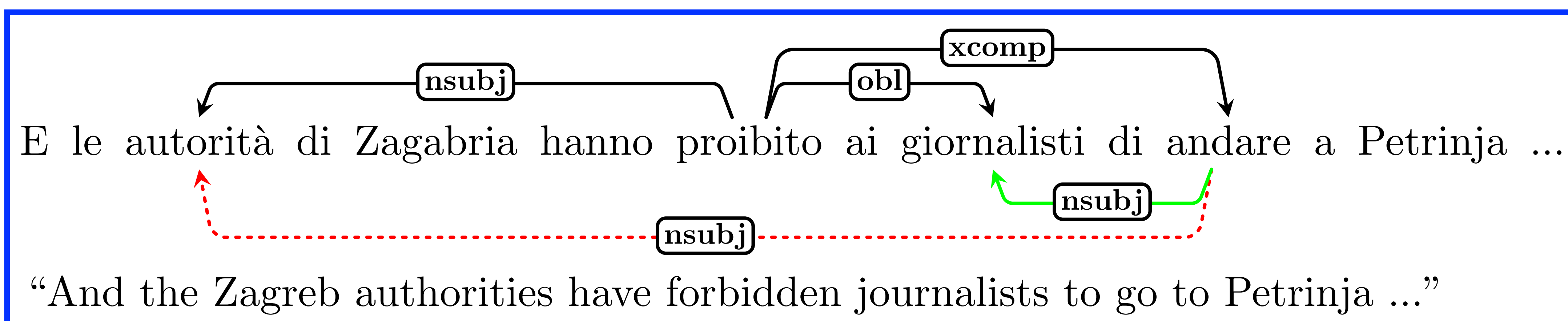
Error Analysis: Subjects

- Stanford has higher recall for both languages because it considers all xcomp nodes (not just infinitives)
- Stanford has higher precision for Swedish because it handles object control



Error Analysis: Subjects

- Stanford has higher recall for both languages because it considers all xcomp nodes (not just infinitives)
- Stanford has higher precision for Swedish because it handles object control
- Stanford has lower precision for Italian because it handles object control (sic)



Error Analysis: Coordination

Error Analysis: Coordination

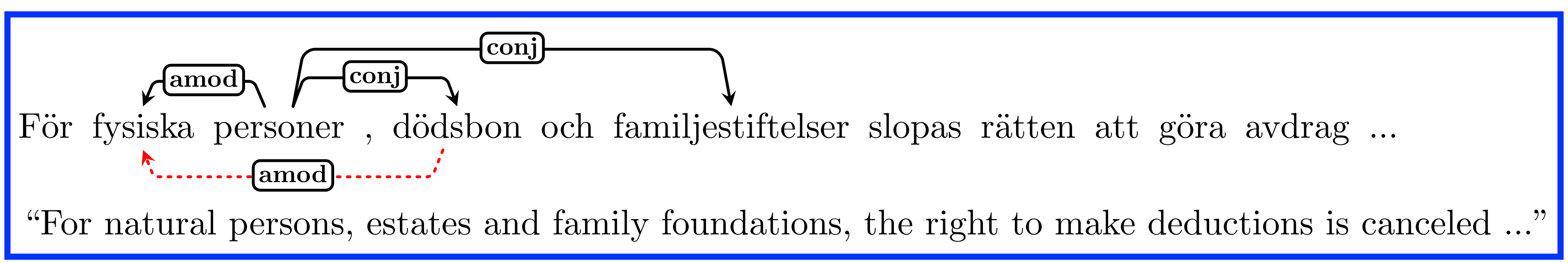
- Stanford has higher precision for both languages because it limits projected dependencies to core arguments

Error Analysis: Coordination

- Stanford has higher precision for both languages because it limits projected dependencies to core arguments
- Turku has higher recall for both languages because it projects all kinds of dependents

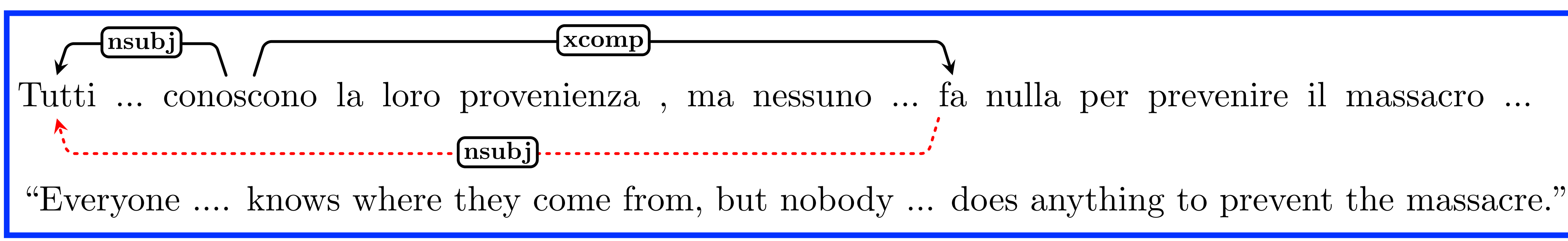
Error Analysis: Coordination

- Stanford has higher precision for both languages because it limits projected dependencies to core arguments
- Turku has higher recall for both languages because it projects all kinds of dependents
- Most common error type (especially for Turku): predicting shared left-dependents of the first conjunct



Error Analysis: Coordination

- Stanford has higher precision for both languages because it limits projected dependencies to core arguments
- Turku has higher recall for both languages because it projects all kinds of dependents
- Most common error type (especially for Turku): predicting shared left-dependents of the first conjunct



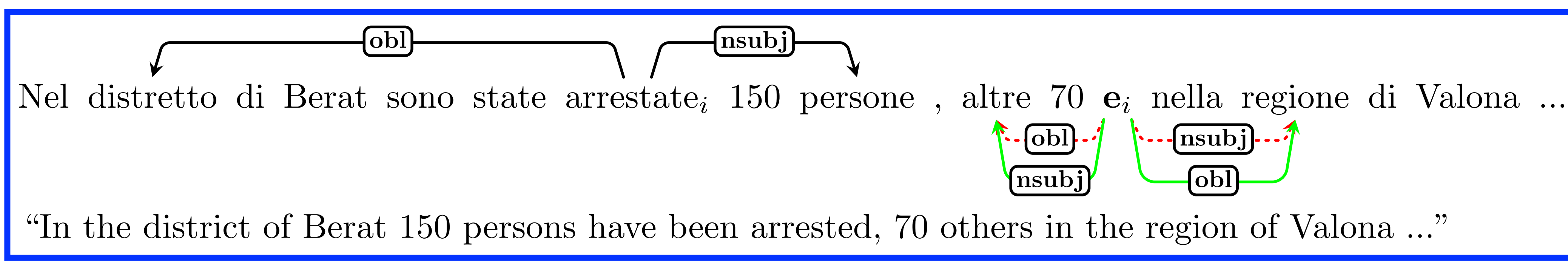
Error Analysis: Ellipsis

Error Analysis: Ellipsis

- The Stanford system developed for English works equally well (or better) for Swedish

Error Analysis: Ellipsis

- The Stanford system developed for English works equally well (or better) for Swedish
- The system has lower precision on Italian due (in part) to different word order constraints



Interim Conclusion

Encouraging results for bootstrapping UD treebanks

- High cross-lingual accuracy – even for unrelated languages
- A few language-specific adaptations may help a lot
- Swedish v2.2 will have enhanced dependencies! 😀🇸🇪

Future work

- Use enhancers to post-process basic dependencies output by parsers

(Some) Open Issues

- Vague guidelines for coordination
- Treatment of semantically void dependents
- Neutralization of syntactic alternations

Coordination

Coordination

- Current guidelines do not specify for which coordinate structures one should add dependencies

Coordination: Dependents

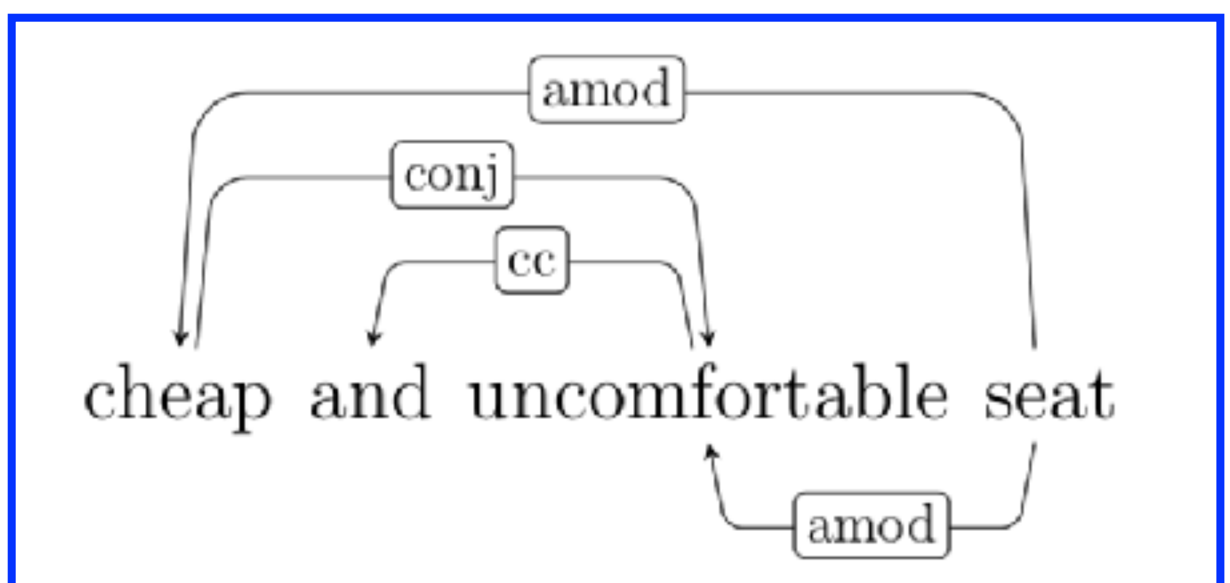
- Current guidelines do not specify for which coordinate structures one should add dependencies
- Case 1 — coordinated dependents: It seems reasonable to always add dependencies

Coordination: Dependents

- Current guidelines do not specify for which coordinate structures one should add dependencies
- Case 1 — coordinated dependents: It seems reasonable to always add dependencies
  - Exception: discourse, parataxis, list, conj, orphan

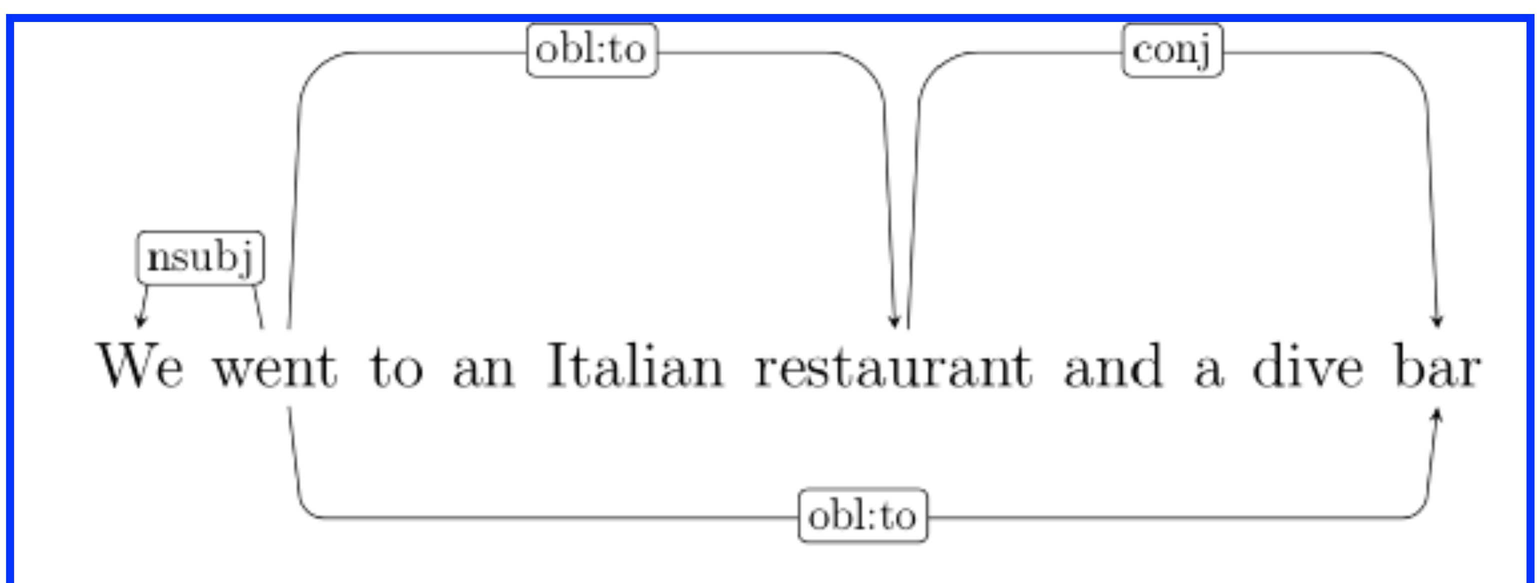
Coordination: Dependents

- Current guidelines do not specify for which coordinate structures one should add dependencies
- Case 1 — coordinated dependents: It seems reasonable to always add dependencies
  - Exception: discourse, parataxis, list, conj, orphan



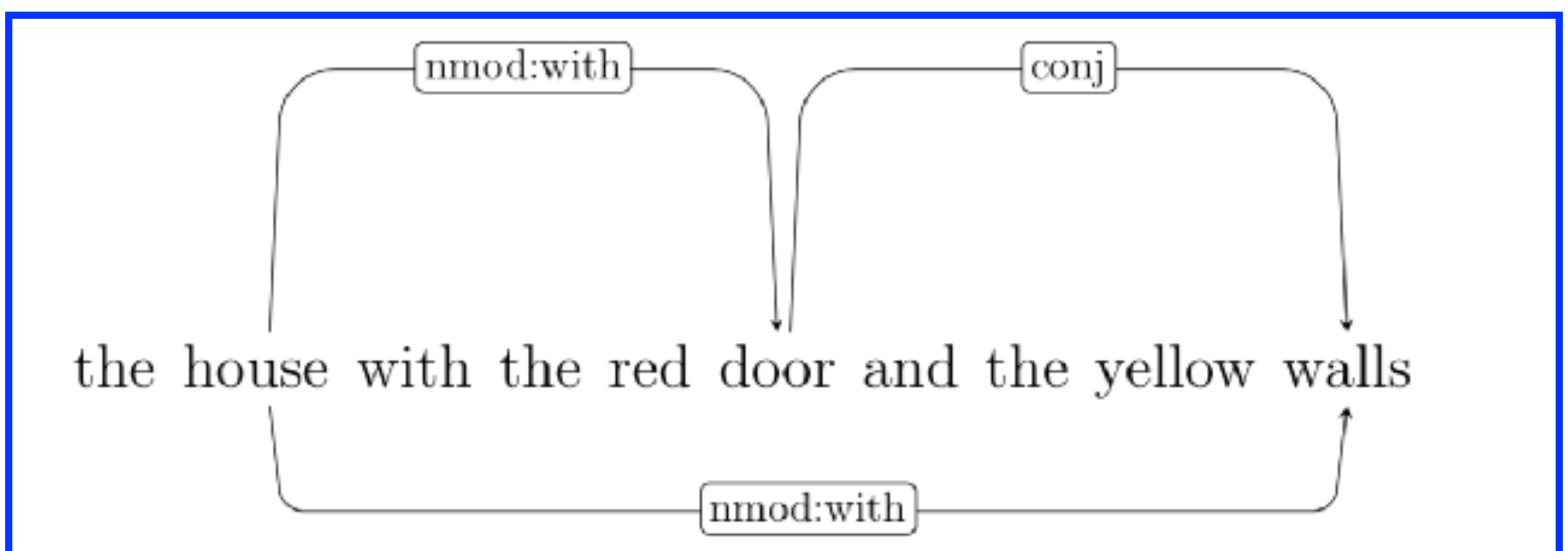
Coordination: Dependents

- Current guidelines do not specify for which coordinate structures one should add dependencies
- Case 1 — coordinated dependents: It seems reasonable to always add dependencies
  - Exception: discourse, parataxis, list, conj, orphan



Coordination: Dependents

- Current guidelines do not specify for which coordinate structures one should add dependencies
- Case 1 — coordinated dependents: It seems reasonable to always add dependencies
  - Exception: discourse, parataxis, list, conj, orphan



Coordination: Dependents

Coordination: Dependents

- This may lead to strange interpretations if one sees this process as a reverse conjunction reduction

Coordination: Dependents

- This may lead to strange interpretations if one sees this process as a reverse conjunction reduction

(1) the black and white movie

Coordination: Dependents

- This may lead to strange interpretations if one sees this process as a reverse conjunction reduction

(1) the black and white movie

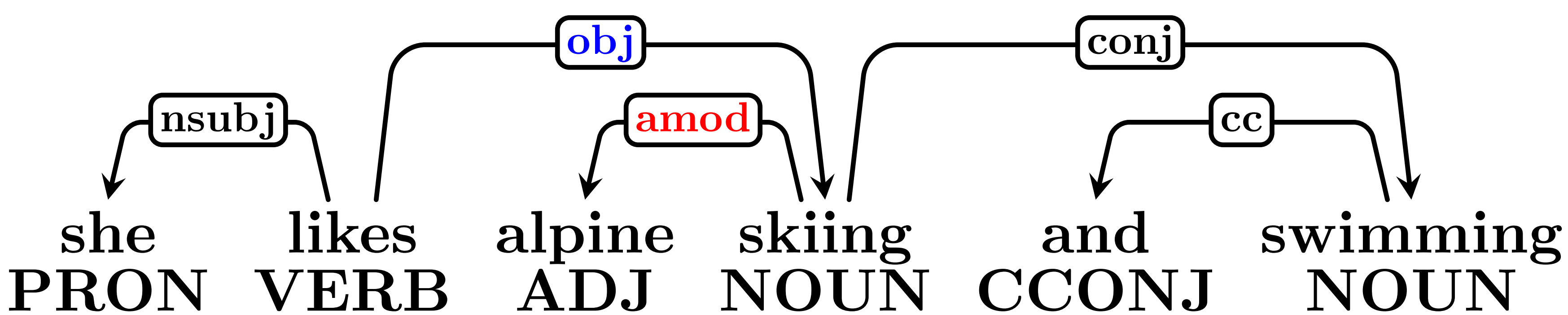
(2) If you put your plate in the dishwasher and take the trash out, you can play outside

Coordination: Heads

- The basic representation does not indicate whether arguments and modifiers are shared across coordinated heads

Coordination: Heads

- The basic representation does not indicate whether arguments and modifiers are shared across coordinated heads



Coordination: Heads

- The basic representation does not indicate whether arguments and modifiers are shared across coordinated heads
- Core arguments (n/csubj, (i)obj, xcomp, ccomp) can appear only once in almost all languages
  - This fact can be used for automatic addition of dependencies

Coordination: Heads

- Non-core arguments/modifiers are less straightforward:

  (1) In the morning, we listened to a talk and we participated in a discussion.

  (2) In the morning, we listened to a talk and an hour ago, we participated in a discussion.

Coordination: Heads

- Evaluation and annotation heuristic:

  Check whether it is syntactically possible to copy shared modifier and whether the semantics is (more or less) preserved.

Coordination: Heads

- Evaluation and annotation heuristic:

  Check whether it is syntactically possible to copy shared modifier and whether the semantics is (more or less) preserved.

  (1) In the morning, we listened to a talk and
  we participated in a discussion.

Coordination: Heads

- Evaluation and annotation heuristic:

  Check whether it is syntactically possible to copy shared modifier and whether the semantics is (more or less) preserved.

  (1) In the morning, we listened to a talk and
  in the morning, we participated in a discussion.

Coordination: Heads

- Evaluation and annotation heuristic:

  Check whether it is syntactically possible to copy shared modifier and whether the semantics is (more or less) preserved.

  (2) In the morning, we listened to a talk and an hour ago, we participated in a discussion.

Coordination: Heads

- Evaluation and annotation heuristic:

  Check whether it is syntactically possible to copy shared modifier and whether the semantics is (more or less) preserved.

  (2) In the morning, we listened to a talk and an hour ago, in the morning, we participated in a discussion.

Coordination: Heads

- Evaluation and annotation heuristic:

  Check whether it is syntactically possible to copy shared modifier and whether the semantics is (more or less) preserved.

  (2) In the morning, we listened to a talk and an hour ago, in the morning, we participated in a discussion.

- Does this heuristic lead to consistent annotations?

Coordination: Heads

- Evaluation and annotation heuristic:

  Check whether it is syntactically possible to copy shared modifier and whether the semantics is (more or less) preserved.

  (2) In the morning, we listened to a talk and an hour ago, in the morning, we participated in a discussion.

- Does this heuristic lead to consistent annotations?
- Can we do this automatically?

Semantically Void Dependents

Semantically Void Dependents

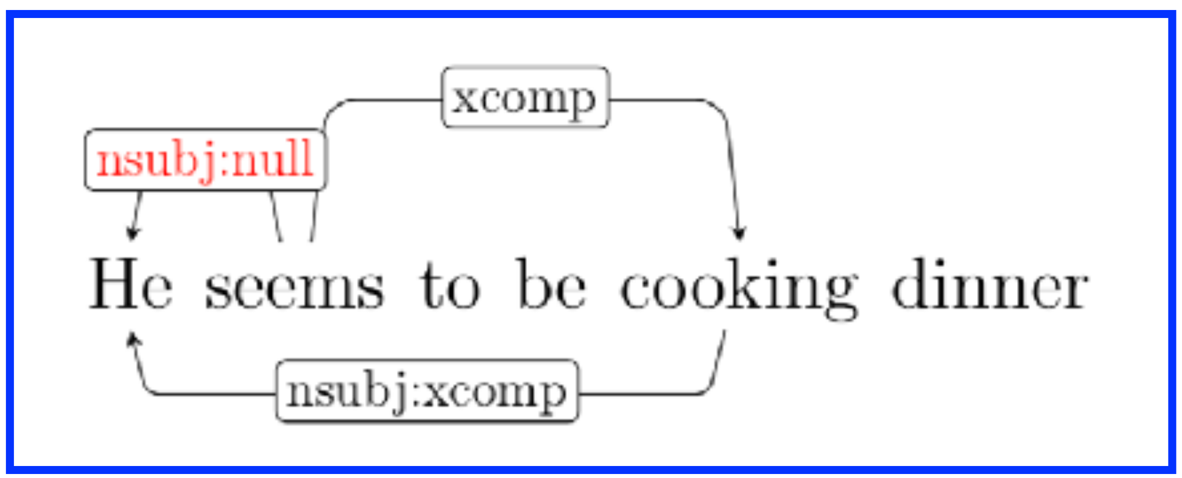
- UD currently does not distinguish between control and raising constructions

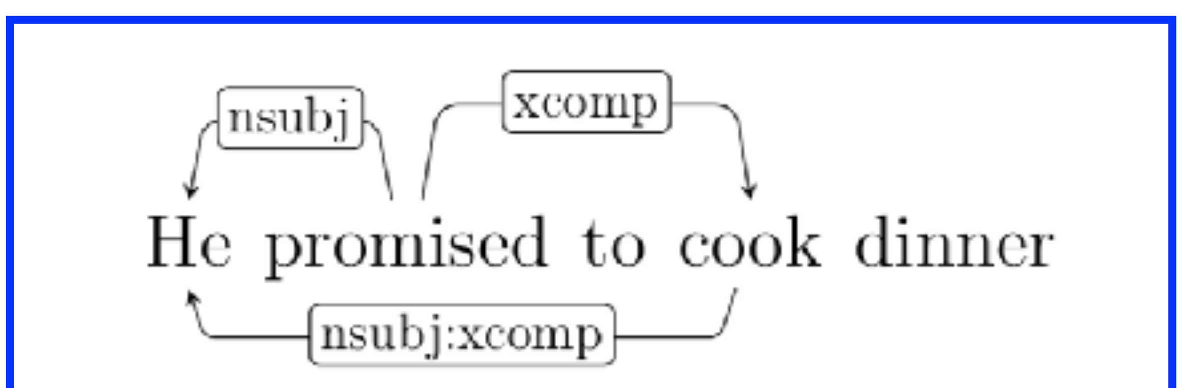
Semantically Void Dependents

- UD currently does not distinguish between control and raising constructions
- Should we mark raised subjects to indicate that they are not a semantic argument of the matrix verb?

Semantically Void Dependents

- UD currently does not distinguish between control and raising constructions
- Should we mark raised subjects to indicate that they are not a semantic argument of the matrix verb?





Syntactic Alternations

Syntactic Alternations

- Syntactic alternations are potentially problematic for semantic downstream tasks

Syntactic Alternations

- Syntactic alternations are potentially problematic for semantic downstream tasks
- Basic (and enhanced) UD mark passives with nsubj:pass relation

Syntactic Alternations

- Syntactic alternations are potentially problematic for semantic downstream tasks
- Basic (and enhanced) UD mark passives with nsubj:pass relation
- Should we actually try to neutralize some syntactic alternations (à la Candito et al. 2017)? Which ones?

Discussion Points

- How to improve the automatic generation of enhanced UD treebanks
- Treatment of coordination in enhanced UD
- Treatment of semantically void arguments in enhanced UD
- Treatment of syntactic alternations in enhanced UD