From dependency structures to LFG representations

Dag Haug

Seminar in computational linguistics
April 18

The texts

- Core – parallel corpus of New Testament translations:
  - Ancient Greek (original, 1st century AD)
  - Gothic (4th century AD)
  - Latin (ca. 400 AD)
  - Classical Armenian (ca. 400 AD)
  - Old Church Slavic (9th century AD)

The texts

- Core – parallel corpus of New Testament translations:
  - Ancient Greek (original, 1st century AD)
  - Gothic (4th century AD)
  - Latin (ca. 400 AD)
  - Classical Armenian (ca. 400 AD)
  - Old Church Slavic (9th century AD)
- Extensions:
  - Herodotus' Histories (Greek 5th century BC)
  - Caesar's Gallic War (Latin, 1st century BC)
  - Cicero's Letters to Atticus (Latin, 1st century BC)
  - Peregrinatio Aetheriae (Vulgar Latin, ca. 400 AD)
  - Hagiographies (The Slavic Codex Suprasliensis, 11th century AD)

The texts

- Core – parallel corpus of New Testament translations:
  - Ancient Greek (original, 1st century AD)
  - Gothic (4th century AD)
  - Latin (ca. 400 AD)
  - Classical Armenian (ca. 400 AD)
  - Old Church Slavic (9th century AD)
- Extensions:
  - Herodotus' Histories (Greek 5th century BC)
  - Caesar's Gallic War (Latin, 1st century BC)
  - Cicero's Letters to Atticus (Latin, 1st century BC)
  - Peregrinatio Aetheriae (Vulgar Latin, ca. 400 AD)
  - Hagiographies (The Slavic Codex Suprasliensis, 11th century AD)
- Ultimate goal: a representative corpus of early IE languages

Small but beautiful

language	tokens
chu	64031
got	56315
grc	137750
lat	120253
xcl	22614
total	400963

On the languages

- Old languages → no native speakers

On the languages

- Old languages → no native speakers
- But fairly well-understood and much-studied texts

On the languages

- Old languages → no native speakers
- But fairly well-understood and much-studied texts
- Morphologically rich

On the languages

- Old languages → no native speakers
- But fairly well-understood and much-studied texts
- Morphologically rich
- Non-configurational, grammatical functions indicated by case rather than word order

On the languages

- Old languages → no native speakers
- But fairly well-understood and much-studied texts
- Morphologically rich
- Non-configurational, grammatical functions indicated by case rather than word order
- All in all quite different from English, which creates lots of problems. . .

Workflow for annotation

- International team of student annotators

Workflow for annotation

- International team of student annotators
- Manual disambiguation of morphology and lemmatization

Workflow for annotation

- International team of student annotators
- Manual disambiguation of morphology and lemmatization
- Syntactic annotation

Workflow for annotation

- International team of student annotators
- Manual disambiguation of morphology and lemmatization
- Syntactic annotation
- Review by project members

Workflow for annotation

- International team of student annotators
- Manual disambiguation of morphology and lemmatization
- Syntactic annotation
- Review by project members
- Advanced annotation done by project members

Morphology

- Verbs inflect for tense, mood, voice, person, number

Morphology

- Verbs inflect for tense, mood, voice, person, number
- Nominals inflect for case, number, gender + possibly grade and definiteness

Morphology

- Verbs inflect for tense, mood, voice, person, number
- Nominals inflect for case, number, gender + possibly grade and definiteness
- All in all this makes for 1817 unique MSD-tags

Morphology

- Verbs inflect for tense, mood, voice, person, number
- Nominals inflect for case, number, gender + possibly grade and definiteness
- All in all this makes for 1817 unique MSD-tags
- In addition there are 25 POS-tags (fairly traditional, with some subdivisions especially in the pronouns)

Morphological annotation

- Started out with manual disambiguation of alternatives from a transducer

Morphological annotation

- Started out with manual disambiguation of alternatives from a transducer
- Ignores the context and offers spurious ambiguities

Morphological annotation

- Started out with manual disambiguation of alternatives from a transducer
- Ignores the context and offers spurious ambiguities
- When we have enough data within a domain, we now use TnT to pretag the text

Morphological annotation

- Started out with manual disambiguation of alternatives from a transducer
- Ignores the context and offers spurious ambiguities
- When we have enough data within a domain, we now use TnT to pretag the text
- MDSs are supplemented with lemmatization from the transducer

Morphological annotation

- Started out with manual disambiguation of alternatives from a transducer
- Ignores the context and offers spurious ambiguities
- When we have enough data within a domain, we now use TnT to pretag the text
- MDSs are supplemented with lemmatization from the transducer
- Skjærholt (2011, 2012):

Experiment	Token accuracy
Cross-validation on BG	84.3%
Vulgate → BG	62.8%

Morphological annotation

- Started out with manual disambiguation of alternatives from a transducer
- Ignores the context and offers spurious ambiguities
- When we have enough data within a domain, we now use TnT to pretag the text
- MDSs are supplemented with lemmatization from the transducer
- Skjærholt (2011, 2012):

Experiment	Token accuracy
Cross-validation on BG	84.3%
Vulgate → BG	62.8%

- Annotation accuracy goes up and time goes down

The syntactic annotation scheme: dependency grammar

- Information about syntactic relations and word order stored separately

The syntactic annotation scheme: dependency grammar

- Information about syntactic relations and word order stored separately
- Reliance on overt elements

The syntactic annotation scheme: dependency grammar

- Information about syntactic relations and word order stored separately
- Reliance on overt elements
- Inherent problems of: (asyndetic) coordination, structure sharing

The syntactic annotation scheme: dependency grammar

- Information about syntactic relations and word order stored separately
- Reliance on overt elements
- Inherent problems of: (asyndetic) coordination, structure sharing
- Dependency grammar with LFG adjustments

The syntactic annotation scheme: dependency grammar

- Information about syntactic relations and word order stored separately
- Reliance on overt elements
- Inherent problems of: (asyndetic) coordination, structure sharing
- Dependency grammar with LFG adjustments
  - Limited set of empty nodes (for asyndetic coordination and ellipsis)

The syntactic annotation scheme: dependency grammar

- Information about syntactic relations and word order stored separately
- Reliance on overt elements
- Inherent problems of: (asyndetic) coordination, structure sharing
- Dependency grammar with LFG adjustments
  - Limited set of empty nodes (for asyndetic coordination and ellipsis)
  - Secondary dependencies (for structure sharing, incl. control/raising)

The syntactic annotation scheme: dependency grammar

- Information about syntactic relations and word order stored separately
- Reliance on overt elements
- Inherent problems of: (asyndetic) coordination, structure sharing
- Dependency grammar with LFG adjustments
  - Limited set of empty nodes (for asyndetic coordination and ellipsis)
  - Secondary dependencies (for structure sharing, incl. control/raising)
  - More granular syntactic relations than usual

Syntactic relations

Label	**Function**
PRED	Predicate
SUB	Subject
OBJ	Object
OBL	Oblique
AG	Agent
ADV	Adverbial
ATR	Attribute
APOS	Apposition
NARG	Nominal argument

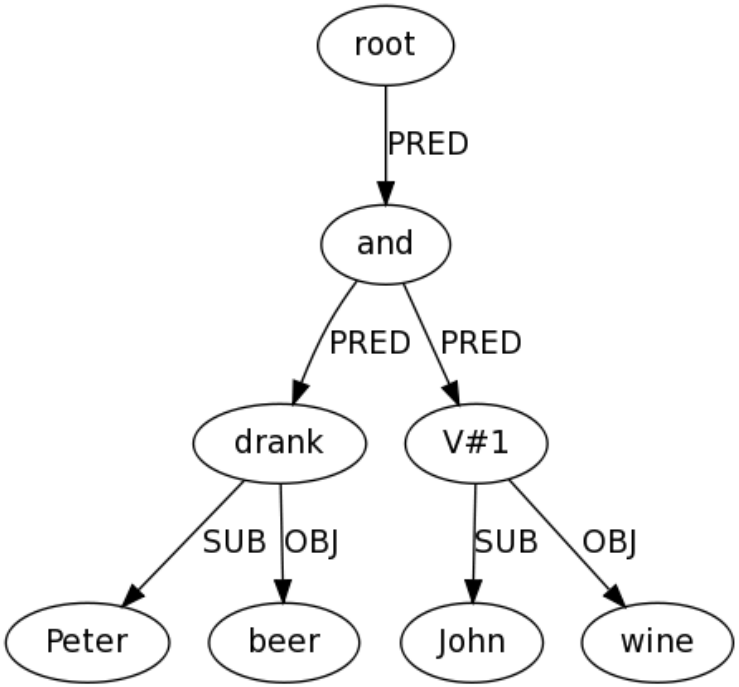
Label	**Function**
XADV	Free predicative
XOBJ	Open complement
Aux	Auxiliary
XOBJ	Open complement clause
COMP	Complement clause
PART	Partitive
PARPRED	Parenthetical
VOC	Vocative

▸ example

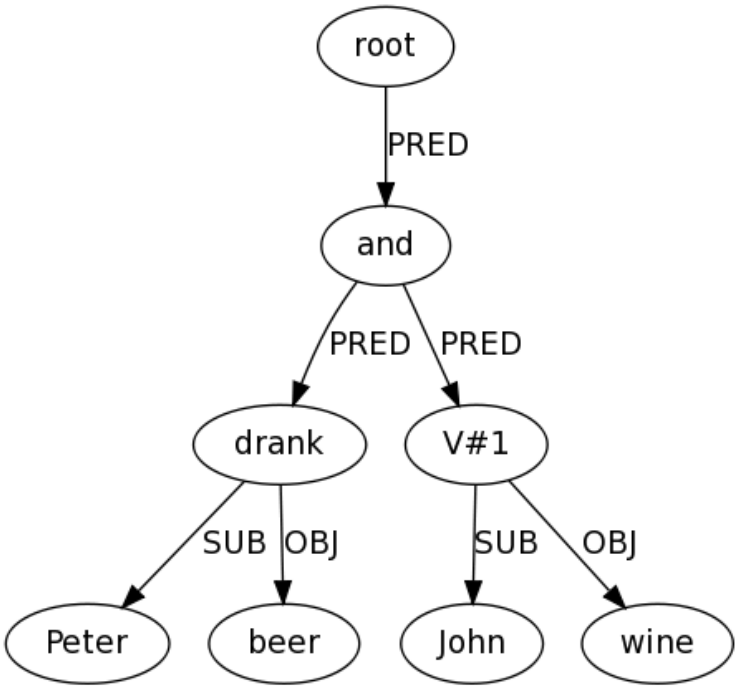
Empty nodes

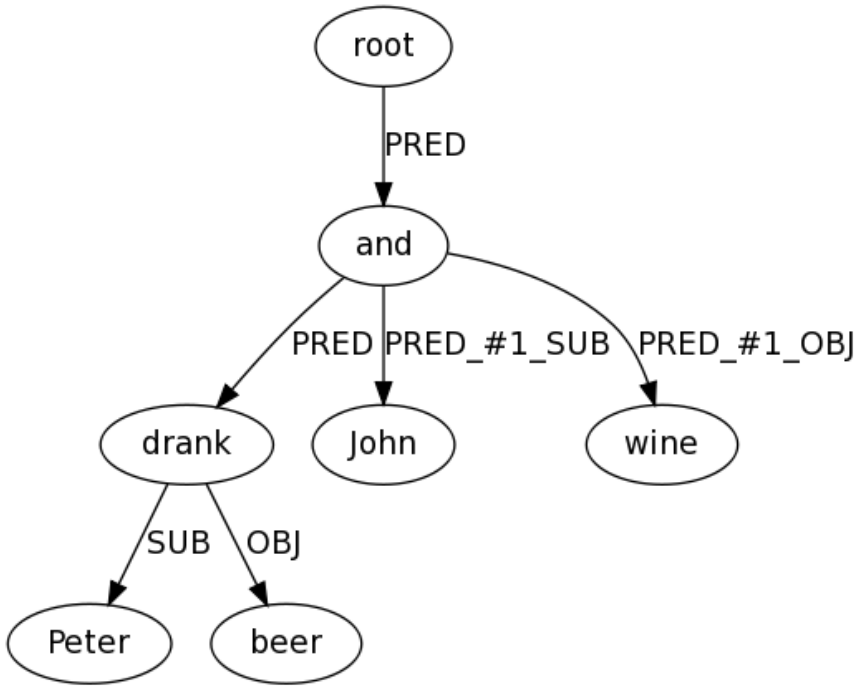
- Null conjunctions for asyndetic parataxis
- Null verbs for null copulas and elided verbs

Eliminability of empty nodes

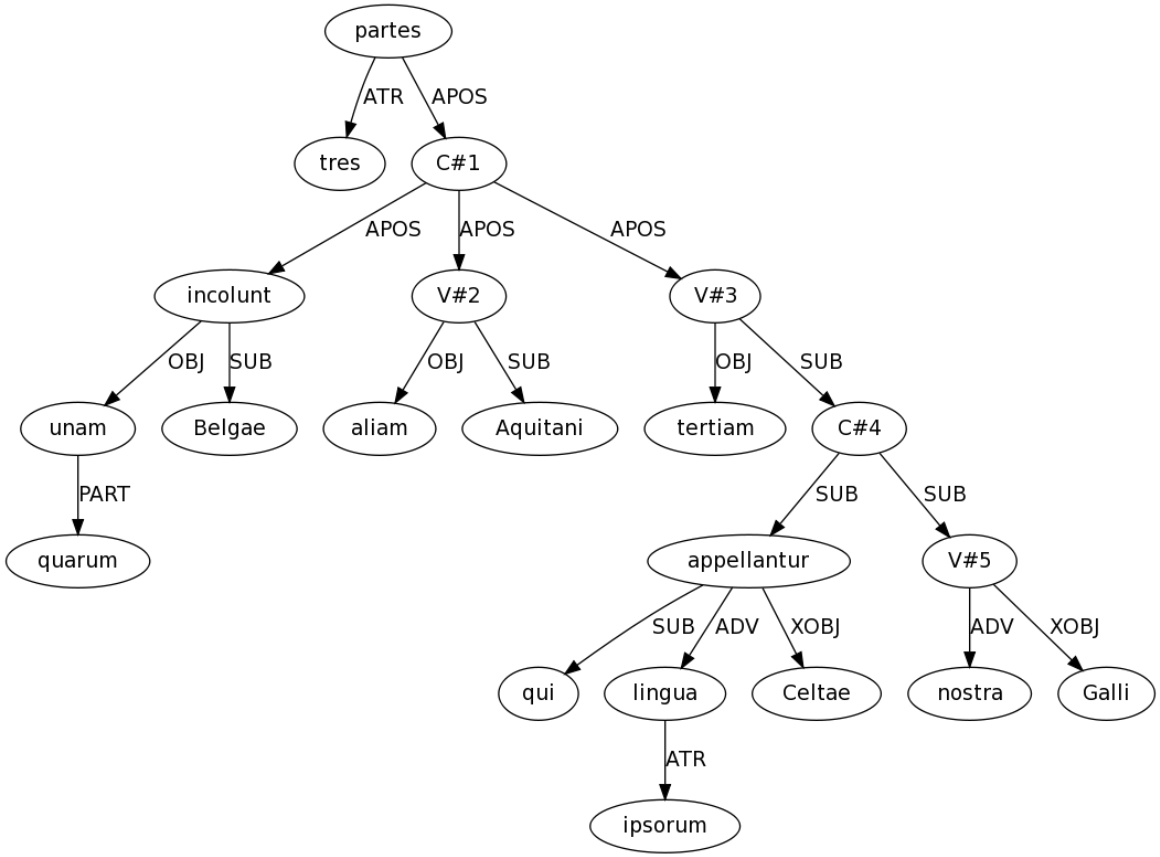


Eliminability of empty nodes



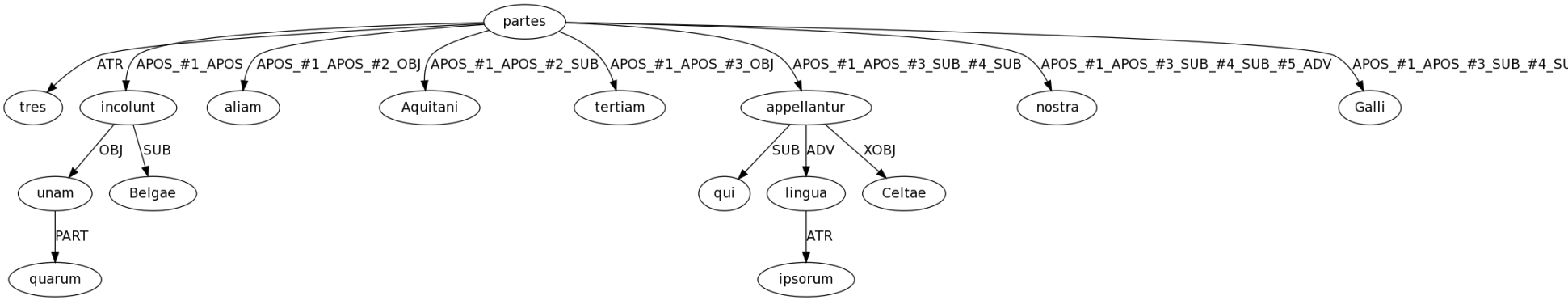


Human processing



of which the Belgians inhabit one, the Aquitani V another,C those who are called Celts in their own language – C Gauls V in our – V the third.

Human processing



of which the Belgians inhabit one, the Aquitani V another,C those who are called Celts in their own language – C Gauls V in our – V the third.

Structure sharing

- Subject control: ▸ Example
- Object control: ▸ Example
- Various other possibilities
- Could also be encoded in the label but typically not with the same precision

Projectivity

language	source	nonprojective	projective
Latin	Gallic War	1887	22717
	Letters to Atticus	2006	20416
	Vulgate	4217	92186
	Per. Aeth.	1279	14890
Greek	Herodotus	6606	56175
	NT	4377	103418
OCS	Zographensis	36	1034
	Suprasliensis	416	7780
	Marianus	1828	47731
Gothic	NT	1886	46884
Armenian	NT	1231	59556
	Koriwn	48	1556

Token alignments

- The translations of the NT have been aligned with the Greek original

Token alignments

- The translations of the NT have been aligned with the Greek original
- A ‘dictionary’ based on likelihood of occurring in the same bible verse

Token alignments

- The translations of the NT have been aligned with the Greek original
- A ‘dictionary’ based on likelihood of occurring in the same bible verse
- Information from the annotation: syntax, morphology, word order

Token alignments

- The translations of the NT have been aligned with the Greek original
- A ‘dictionary’ based on likelihood of occurring in the same bible verse
- Information from the annotation: syntax, morphology, word order
- Manual correction of the Slavic indicates very good results (and a very literal translation)

Precision	Recall	F-score
95.27%	92.97%	94.11%

Givenness

- Givenness tags based on which context the hearer uses to establish reference
  - Discourse (anaphora) $\rightarrow$ OLD

Givenness

- Givenness tags based on which context the hearer uses to establish reference
  - Discourse (anaphora) → OLD
  - Situation (deixis) → ACC-SIT

Givenness

- Givenness tags based on which context the hearer uses to establish reference
  - Discourse (anaphora) → OLD
  - Situation (deixis) → ACC-SIT
  - Scenarios (inferences) → ACC-INF

Givenness

- Givenness tags based on which context the hearer uses to establish reference
  - Discourse (anaphora) → OLD
  - Situation (deixis) → ACC-SIT
  - Scenarios (inferences) → ACC-INF
  - Encyclopedic knowledge → ACC-GEN

Givenness

- Givenness tags based on which context the hearer uses to establish reference
  - Discourse (anaphora) → OLD
  - Situation (deixis) → ACC-SIT
  - Scenarios (inferences) → ACC-INF
  - Encyclopedic knowledge → ACC-GEN
  - No context (no extra-NP information) → NEW

Givenness

- Givenness tags based on which context the hearer uses to establish reference
  - Discourse (anaphora) → OLD
  - Situation (deixis) → ACC-SIT
  - Scenarios (inferences) → ACC-INF
  - Encyclopedic knowledge → ACC-GEN
  - No context (no extra-NP information) → NEW

▸ example

Modal subordination

Luke 5:39

Und niemand ist, der vom alten trinkt und wolle bald den neuen; denn er spricht: Der alte ist milder.

- The subject and the old and the new wine are embedded under subordination
- Should be inaccessible (Karttunen, COLING 69) but they aren't
- We ignore recursive embeddings and use a special tagset for all embedded referents

Tagset for embedded referents

- NONSPEC (but QUANT for quantification)
- NONSPEC_INF
- NONSPEC_OLD

Tagset for embedded referents

- NONSPEC (but QUANT for quantification)
- NONSPEC_INF
- NONSPEC_OLD

No counterparts to ACC-GEN or ACC-SIT as these belong in the main DRS by definition

Interannotator agreement

- Towards the end of the NT tagging projects, kappa values were around 0.8 (after long periods of weekly meetings)

Interannotator agreement

- Towards the end of the NT tagging projects, kappa values were around 0.8 (after long periods of weekly meetings)
- New project: Caesar's Gallic War

Interannotator agreement

- Towards the end of the NT tagging projects, kappa values were around 0.8 (after long periods of weekly meetings)
- New project: Caesar's Gallic War
- Supervised tagging of 8 chapters (ca. 400 taggables)

Interannotator agreement

- Towards the end of the NT tagging projects, kappa values were around 0.8 (after long periods of weekly meetings)
- New project: Caesar's Gallic War
- Supervised tagging of 8 chapters (ca. 400 taggables)
- Unsupervised tagging of 5 chapters (ca. 250 taggables)
  - $\kappa = 0.66$ counting divergences in taggables
  - $\kappa = 0.75$ on tags set by both annotators

Interannotator agreement

- Towards the end of the NT tagging projects, kappa values were around 0.8 (after long periods of weekly meetings)
- New project: Caesar's Gallic War
- Supervised tagging of 8 chapters (ca. 400 taggables)
- Unsupervised tagging of 5 chapters (ca. 250 taggables)
  - $\kappa = 0.66$ counting divergences in taggables
  - $\kappa = 0.75$ on tags set by both annotators
- Decent; but much potential for more agreement, especially in taggables

Size of IS corpus

Tag	Freq
old	34430
old_inact	1395
acc_gen	3755
acc_inf	2634
acc_sit	883
new	5768
kind	1178
non_spec	4485
non_spec_inf	408
non_spec_old	1799
quant	2021
total	58756

edge type	freq
coreference	36650
bridging	2847
total	39497

Storing linguistic analyses

- Theory-neutrality →
  - data for larger audiences

Storing linguistic analyses

- Theory-neutrality →
  - data for larger audiences
  - widening gulf between corpus linguistics and linguistic theory

Storing linguistic analyses

- Theory-neutrality →
  - data for larger audiences
  - widening gulf between corpus linguistics and linguistic theory
- DG corpora (Prague, PROIEL) → DG not really in use as a linguistic theory

Storing linguistic analyses

- Theory-neutrality →
  - data for larger audiences
  - widening gulf between corpus linguistics and linguistic theory
- DG corpora (Prague, PROIEL) → DG not really in use as a linguistic theory
- PS corpora (Penn, NEGRA) typically use flatter tree structures than anyone believes in

Storing linguistic analyses

- Theory-neutrality →
  - data for larger audiences
  - widening gulf between corpus linguistics and linguistic theory
- DG corpora (Prague, PROIEL) → DG not really in use as a linguistic theory
- PS corpora (Penn, NEGRA) typically use flatter tree structures than anyone believes in
- On the other hand, LFG and HPSG corpora can be hard to use for people who do not share the theoretical assumptions of these theories

Our take

Principles

1. Encode no more structure than is common to all frameworks

Our take

Principles

1. Encode no more structure than is common to all frameworks
2. Enoded structure could be seen as derived/secondary in some frameworks

Our take

Principles

1. Encode no more structure than is common to all frameworks
2. Enoded structure could be seen as derived/secondary in some frameworks
3. Encode enough structure to allow reconstruction of theoretically motived structures

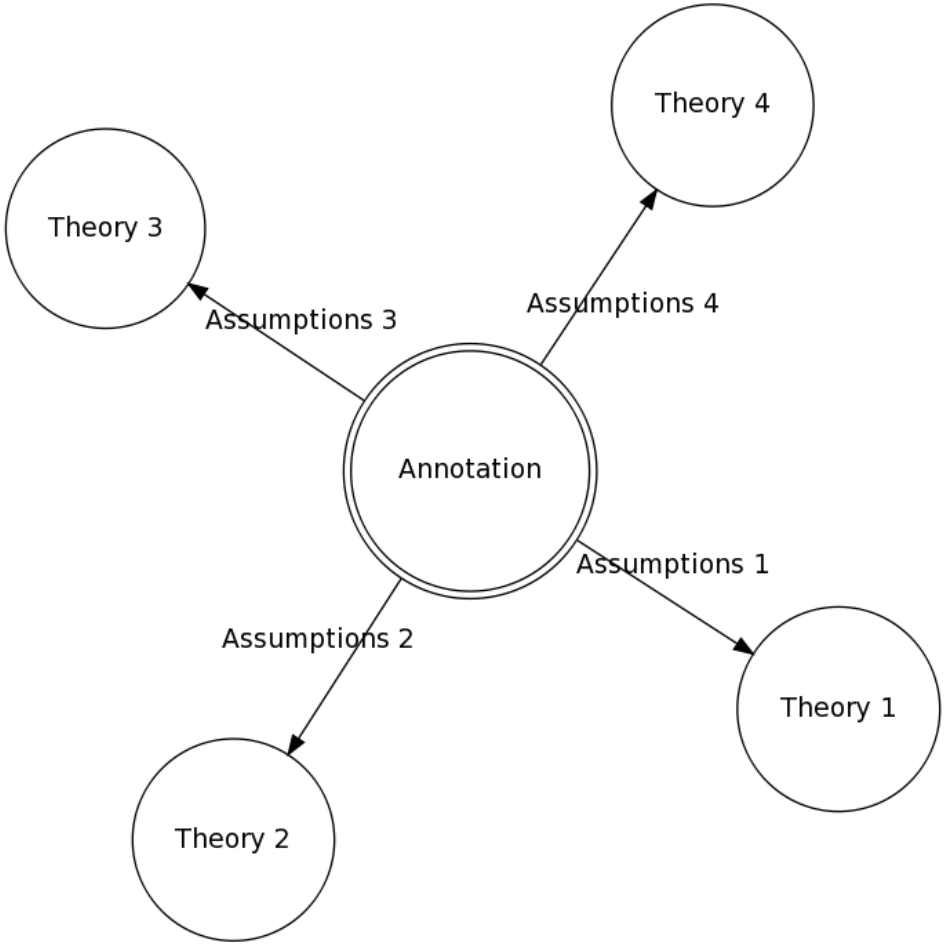
Our take

Principles

1. Encode no more structure than is common to all frameworks
2. Enoded structure could be seen as derived/secondary in some frameworks
3. Encode enough structure to allow reconstruction of theoretically motived structures

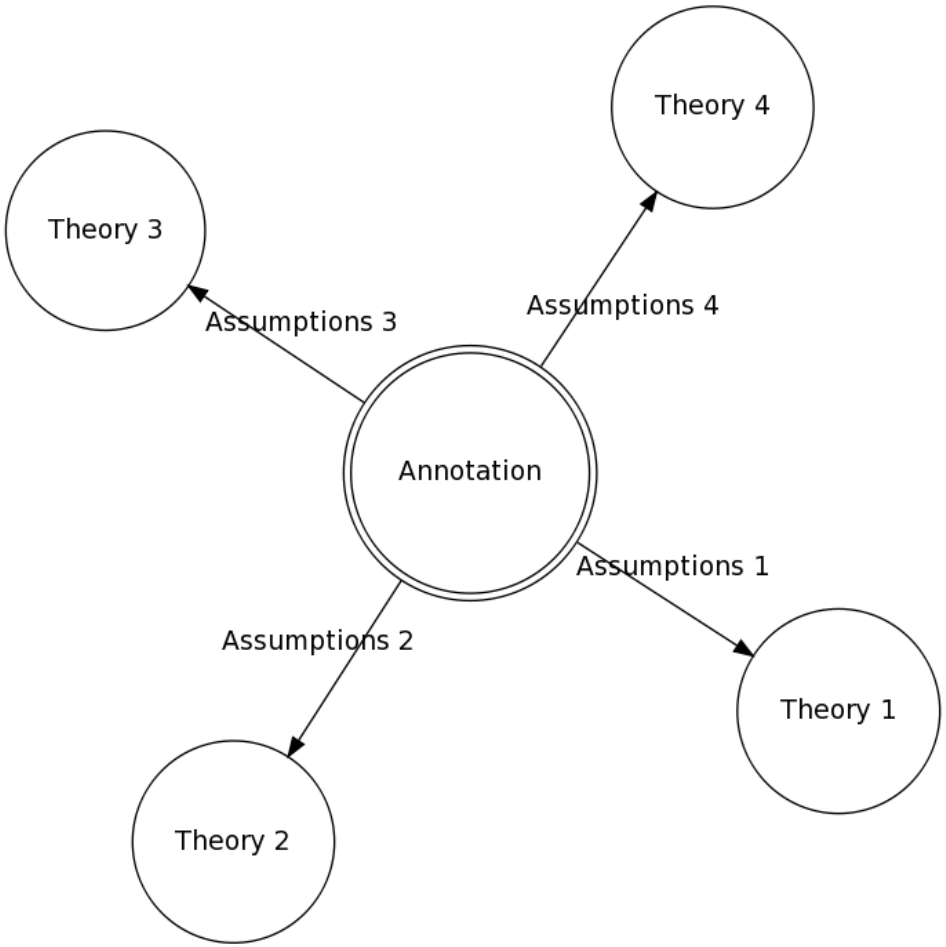
- In the ideal situation, the information in the annotation can be (monotonically) expanded to structures conforming to a particular theory by adding information from the assumptions of that theory

The ideal situation



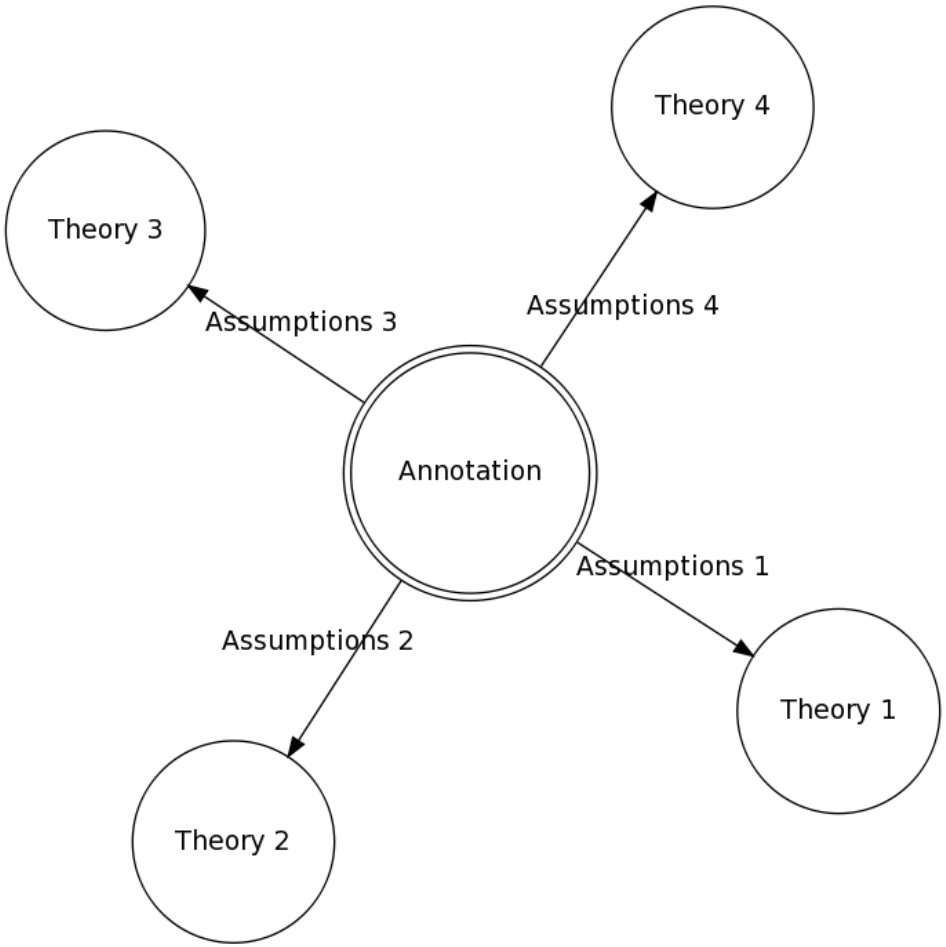
- The added assumptions will typically be about phrase structure, such as various versions of X′ theory

The ideal situation



- The added assumptions will typically be about phrase structure, such as various versions of X′ theory
- Given information about what the subject is, it will be possible to create a structure where the subject has a specific position if the theory requires that (unless the data contradict the theory)

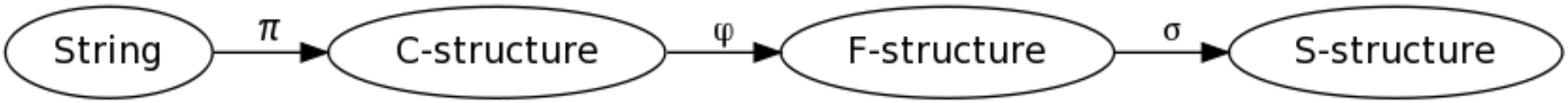
The ideal situation



- The added assumptions will typically be about phrase structure, such as various versions of X′ theory
- Given information about what the subject is, it will be possible to create a structure where the subject has a specific position if the theory requires that (unless the data contradict the theory)
- Useful for hypothesis testing

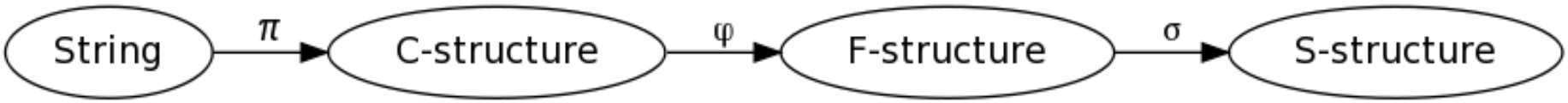
Basic principles

- Modular: several levels of grammatical description connected by projections (functions)



Basic principles

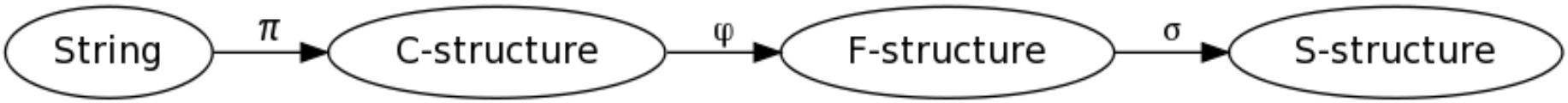
- Modular: several levels of grammatical description connected by projections (functions)



- The c-structure is a tree structure described by a CFG

Basic principles

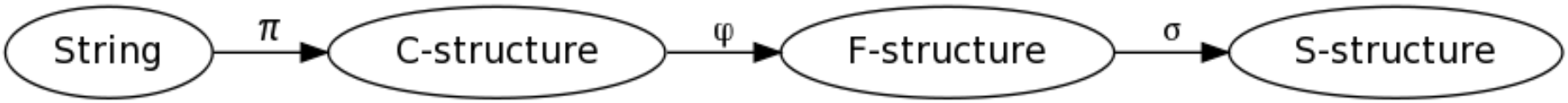
- Modular: several levels of grammatical description connected by projections (functions)



- The c-structure is a tree structure described by a CFG
- The f-structure is a set of ordered attribute-value pairs

Basic principles

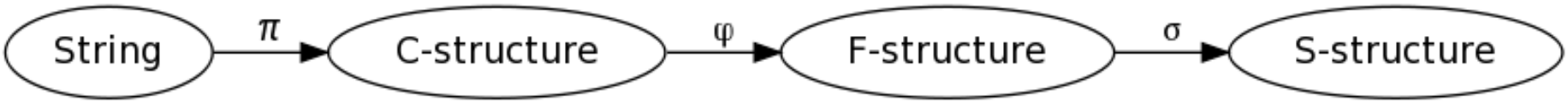
- Modular: several levels of grammatical description connected by projections (functions)



- The c-structure is a tree structure described by a CFG
- The f-structure is a set of ordered attribute-value pairs
- the attribute is a grammatical function or feature and the value is
  - a symbol
  - a semantic form
  - an f-structure
  - a set of f-structures (for adjuncts)

Basic principles

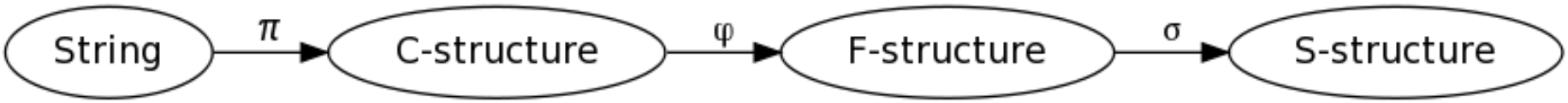
- Modular: several levels of grammatical description connected by projections (functions)



- The c-structure is a tree structure described by a CFG
- The f-structure is a set of ordered attribute-value pairs
- the attribute is a grammatical function or feature and the value is
  - a symbol
  - a semantic form
  - an f-structure
  - a set of f-structures (for adjuncts)
- Lexical items and CFG rules can contribute f-descriptions

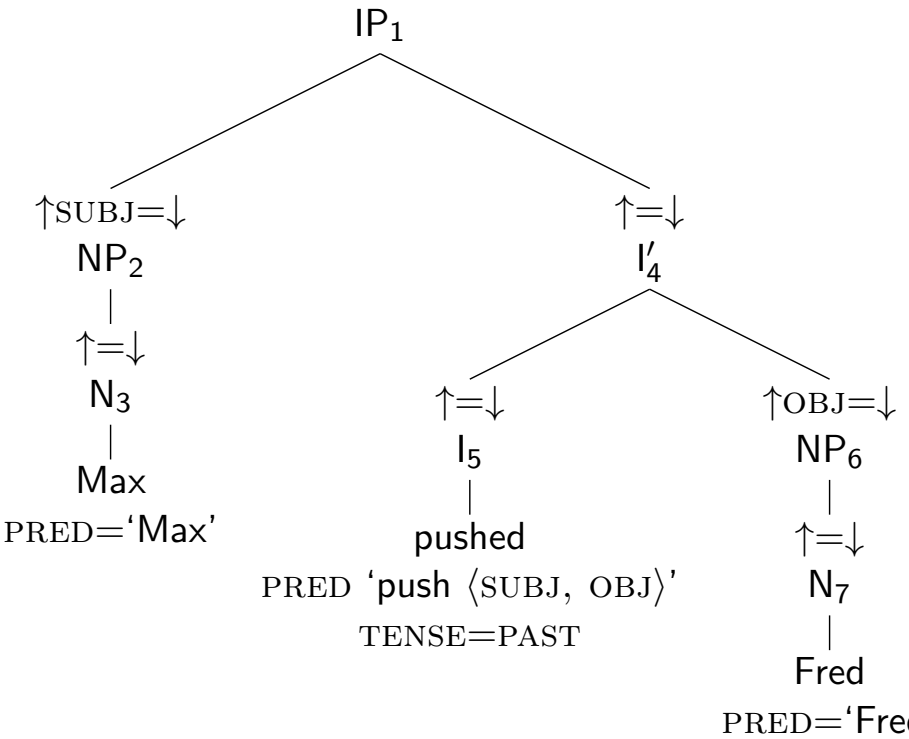
Basic principles

- Modular: several levels of grammatical description connected by projections (functions)



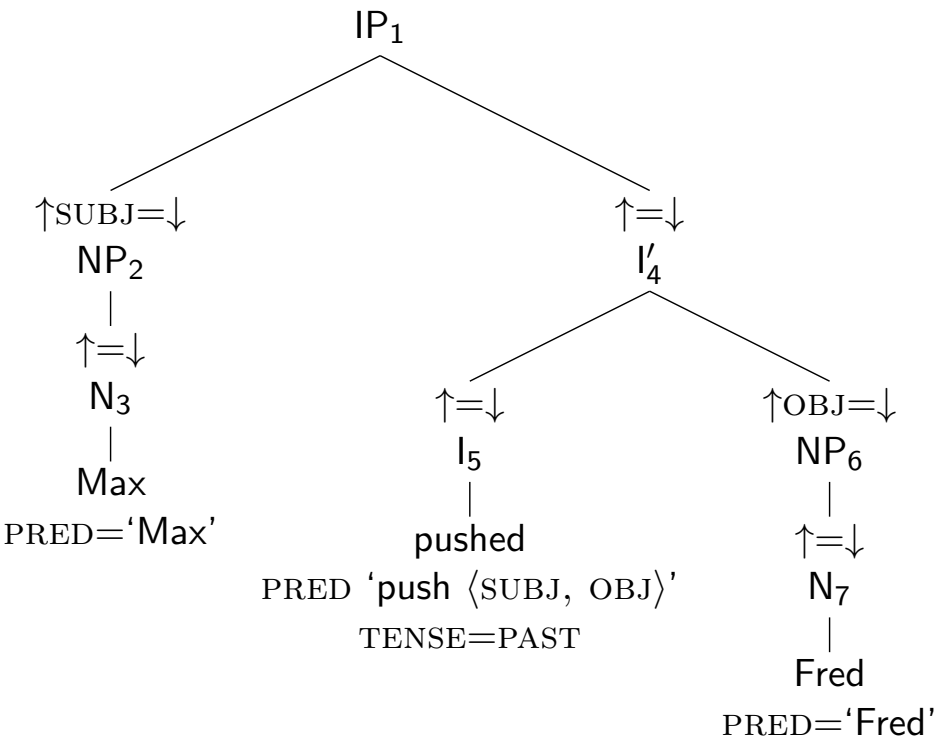
- The c-structure is a tree structure described by a CFG
- The f-structure is a set of ordered attribute-value pairs
- the attribute is a grammatical function or feature and the value is
  - a symbol
  - a semantic form
  - an f-structure
  - a set of f-structures (for adjuncts)
- Lexical items and CFG rules can contribute f-descriptions
- Lexical-functional languages $\in$ context-sensitive languages

Configurational encoding



- 1 SUBJ = 2
- 2 = 3
- 1 = 4
- 4 = 5
- 4 OBJ = 6
- 6 = 7

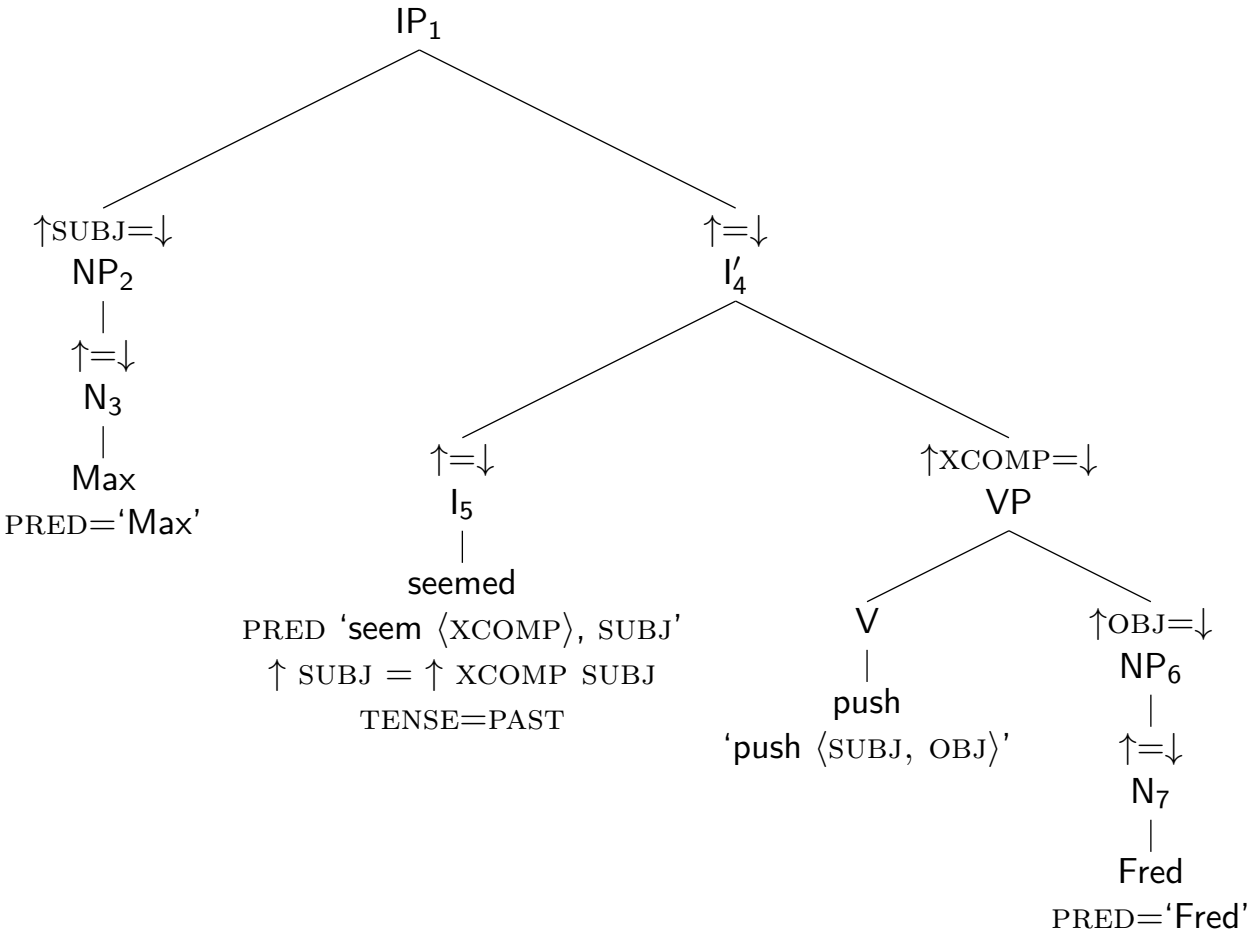
Configurational encoding



1,4,5

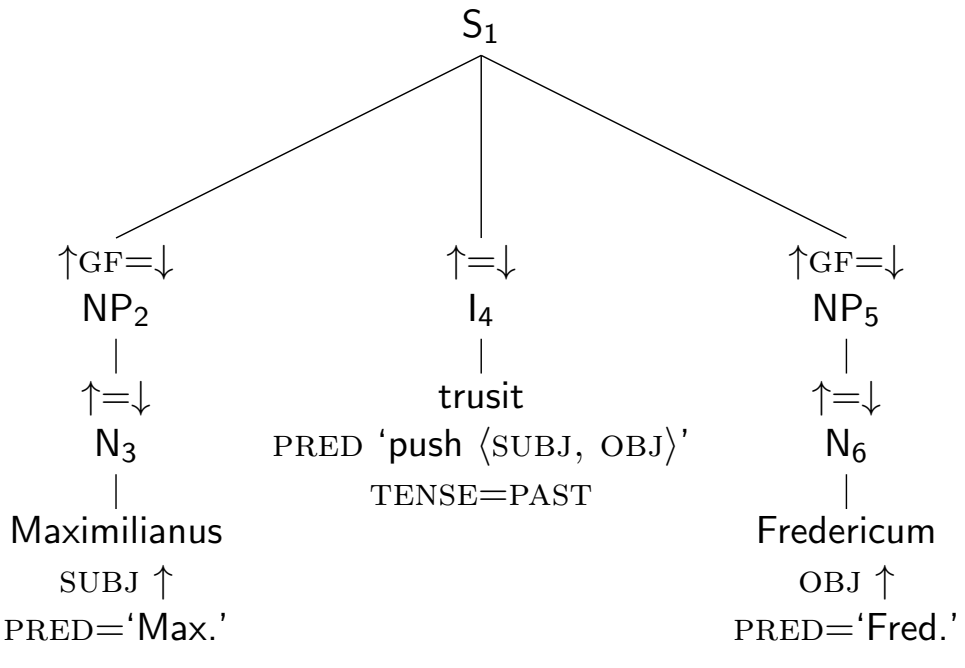
$$\begin{bmatrix} \text{PRED} & \text{`PUSH}\left\langle \text{SUBJ, OBJ} \right\rangle\text{'} \\ \text{SUBJ} & {\scriptstyle 2,3}\begin{bmatrix} \text{PRED} & \text{`MAX'} \end{bmatrix} \\ \text{OBJ} & {\scriptstyle 6,7}\begin{bmatrix} \text{PRED} & \text{`FRED'} \end{bmatrix} \\ \text{TENSE} & \text{PAST} \end{bmatrix}$$

Structure sharing



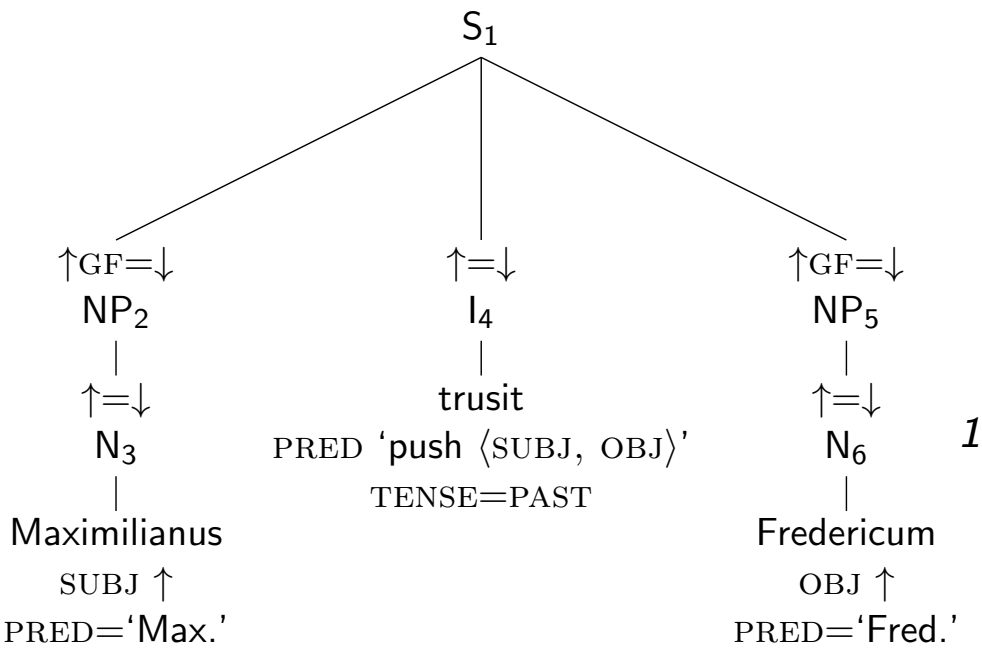
$$\begin{bmatrix} \text{PRED} & \text{'SEEM } \langle \text{XCOMP} \rangle, \text{ SUBJ'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{'MAX'} \end{bmatrix} \\ \text{XCOMP} & \begin{bmatrix} \text{PRED} & \text{'PUSH } \langle \text{SUBJ, OBJ} \rangle \text{'} \\ \text{SUBJ} & — \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{'FRED'} \end{bmatrix} \end{bmatrix} \\ \text{TENSE} & \text{PAST} \end{bmatrix}$$

Non-configurational encoding



- 1 GF = 2
- 2 = 3
- $\exists f.f$ SUBJ = 3
- 1 = 4
- 4 GF = 5
- 5 = 6
- $\exists f.f$ OBJ = 6

Non-configurational encoding



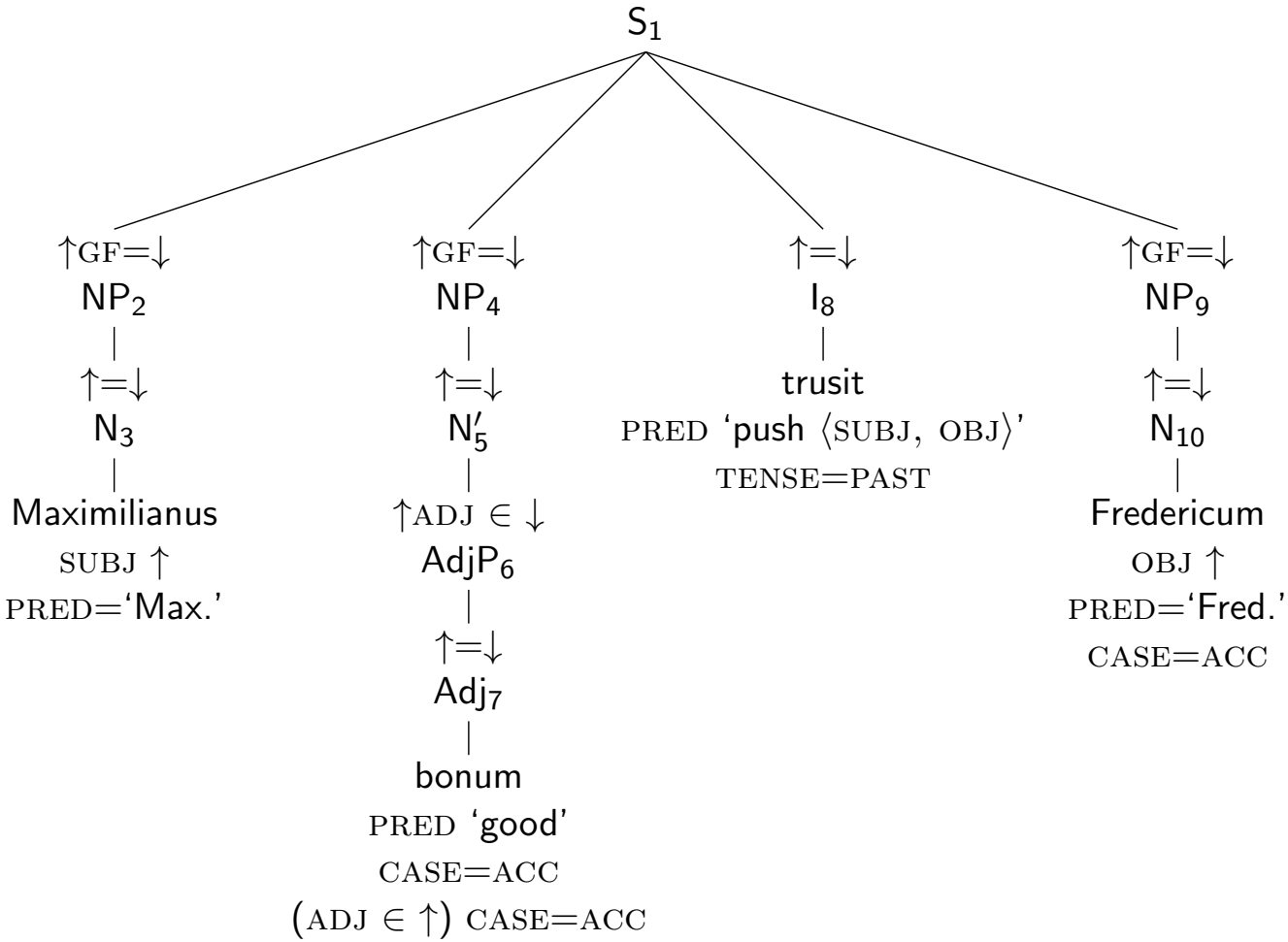
$$\overset{1,4,5}{\begin{bmatrix} \text{PRED} & \text{`PUSH}\left\langle \text{SUBJ, OBJ}\right\rangle\text{'} \\ \text{SUBJ} & \overset{2,3}{\begin{bmatrix} \text{PRED} & \text{`MAX.'} \end{bmatrix}} \\ \text{OBJ} & \overset{6,7}{\begin{bmatrix} \text{PRED} & \text{`FRED.'} \end{bmatrix}} \\ \text{TENSE} & \text{PAST} \end{bmatrix}}$$

Non-projectivity

A mock Latin example

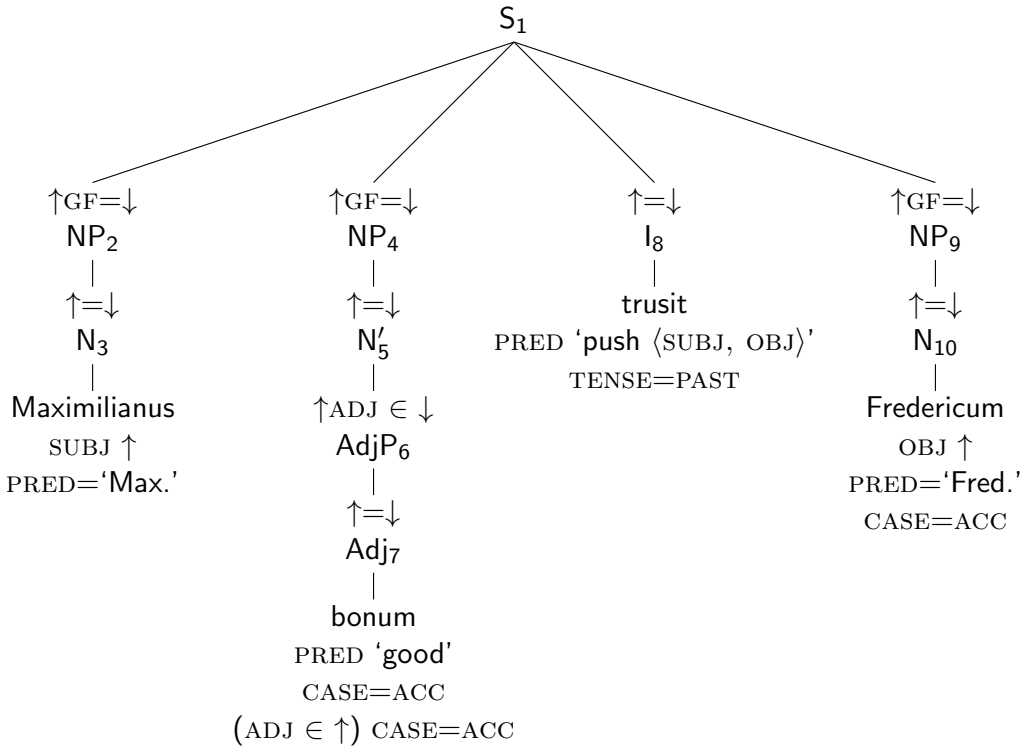
Maximilianus	bonum	trusit	Fredericum
Maximilian.NOM	good.ACC	pushed	Frederick.ACC

Non-projectivity



- 1 GF = 2
- 2 = 3
- $\exists f.f$ SUBJ = 3
- 1 GF = 4
- 4 = 5
- 6 ∈ 5 ADJ
- 7 = 6
- 7 CASE = ACC
- 5 CASE = ACC
- 1 = 8
- 1 GF = 9
- 9 = 10
- 5 CASE = ACC
- $\exists f.f$ OBJ = 10

Non-projectivity



$$
\begin{bmatrix}
\text{PRED} & \text{'PUSH}\left\langle \text{SUBJ, OBJ}\right\rangle\text{'} \\
\text{SUBJ} & {}_{2,3}\begin{bmatrix}\text{PRED} & \text{'MAX.'}\end{bmatrix} \\
\text{OBJ} & {}_{4,5,9,10}\begin{bmatrix}\text{PRED} & \text{'FRED.'} \\ \text{CASE} & \text{ACC} \\ \text{ADJ} & \left\{ {}_{6,7}\begin{bmatrix}\text{PRED} & \text{'GOOD'} \\ \text{CASE} & \text{ACC}\end{bmatrix} \right\}\end{bmatrix} \\
\text{TENSE} & {}_{1,8}\text{PAST}
\end{bmatrix}
$$

Relationship to DG

- F-structures and DGs both encode labelled syntactic dependencies

Relationship to DG

- F-structures and DGs both encode labelled syntactic dependencies
- Two major differences
  - LFG's structure sharing runs against DG's unique head principle
  - In DG, every word introduces depth in the graph, whereas multiple words can contribute to the same F-structure (without nesting)

Relationship to DG

- F-structures and DGs both encode labelled syntactic dependencies
- Two major differences
  - LFG's structure sharing runs against DG's unique head principle
  - In DG, every word introduces depth in the graph, whereas multiple words can contribute to the same F-structure (without nesting)
- We have already given up the unique head principle in our DG

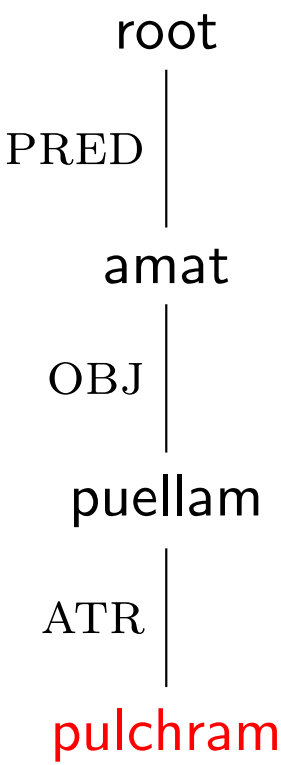
Relationship to DG

- F-structures and DGs both encode labelled syntactic dependencies
- Two major differences
  - LFG's structure sharing runs against DG's unique head principle
  - In DG, every word introduces depth in the graph, whereas multiple words can contribute to the same F-structure (without nesting)
- We have already given up the unique head principle in our DG
- The words that do not introduce separate layers of f-structures are typically function words, so they can be identified from the labels

Label mapping

Function	Label	LFG	Function	Label	LFG
Adverbial	ADV	ADJ	Oblique	OBL	OBJ_θ/OBL
Agent	AG	OBL_{AG}	Parenthetical	PARPRED	—
Apposition	APOS	ADJ	Partitive	PART	ADJ
Attribute	ATR	ADJ	Predicate	PRED	—
Auxiliary	AUX	—	Subject	SUB	SUBJ
Complement	COMP	COMP	Vocative	VOC	—
Argument of noun	NARG	≈ OBL	Free predicative	XADV	XADJ
Object	OBJ	OBJ	Open complement	XOBJ	XCOMP

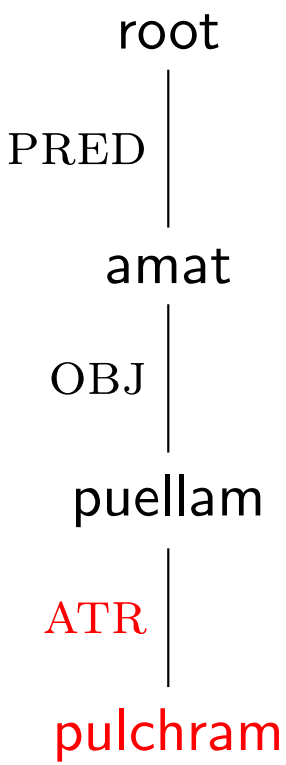
A simple example



- Each node maps to an attribute-value matrix with morphological features and a semantic form

$$\begin{bmatrix} \text{PRED} & \text{'PULCHER'} \\ \text{CASE} & \text{ACC} \\ \text{GEND} & \text{FEM} \end{bmatrix}$$

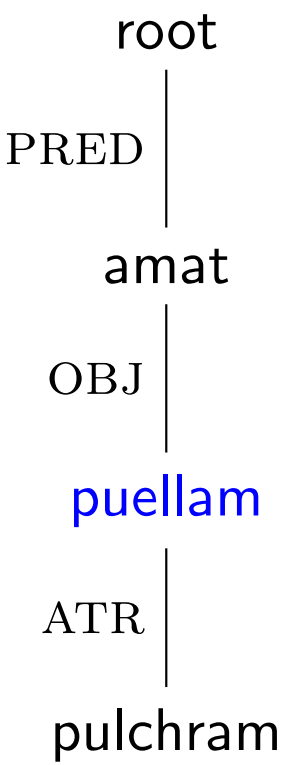
A simple example



- The relations are translated to attributes with the dependents' AVM as value

$$\begin{bmatrix} \text{ADJ} & \left\{ \begin{bmatrix} \text{PRED} & \text{'PULCHER'} \\ \text{CASE} & \text{ACC} \\ \text{GEND} & \text{FEM} \end{bmatrix} \right\} \end{bmatrix}$$

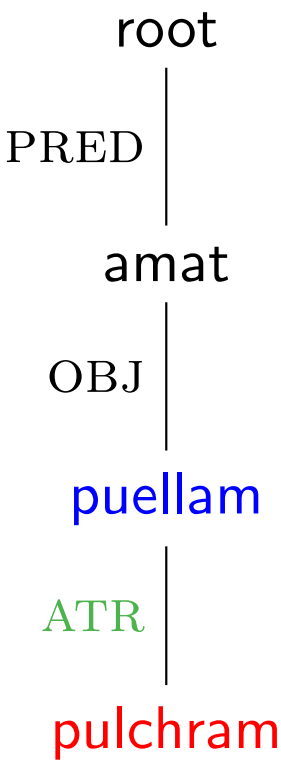
A simple example



- We do this for all nodes in the structure

$$\begin{bmatrix} \text{PRED} & \text{'PUELLA'} \\ \text{CASE} & \text{ACC} \\ \text{GEND} & \text{FEM} \end{bmatrix}$$

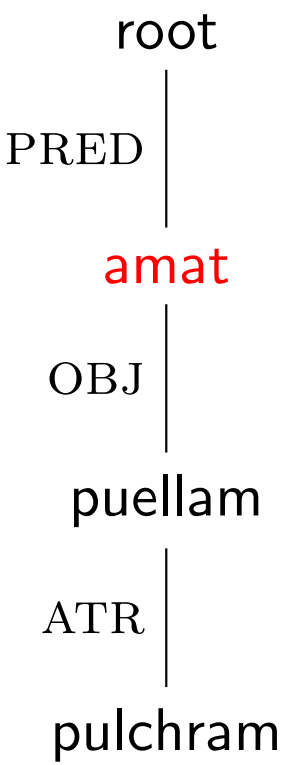
A simple example



- The AVMs of the head and the relation+dependent are unified

$$\begin{bmatrix} \text{PRED} & \text{'PUELLA'} \\ \text{CASE} & \text{ACC} \\ \text{GEND} & \text{FEM} \\ \text{ADJ} & \left\{ \begin{bmatrix} \text{PRED} & \text{'PULCHER'} \\ \text{CASE} & \text{ACC} \\ \text{GEND} & \text{FEM} \end{bmatrix} \right\} \end{bmatrix}$$

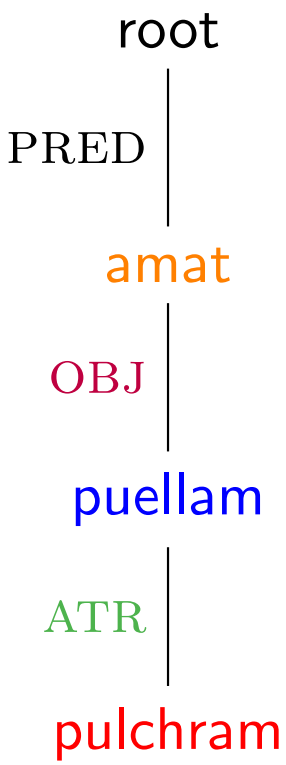
A simple example



- The process terminates with the main verb
- NB $\textsc{pred} \neq \textsc{pred}$

$$\left[\textsc{pred} \quad \text{'}\textsc{amare}\ \langle\ \textsc{subj},\ \textsc{obj}\ \rangle\text{'}\right]$$

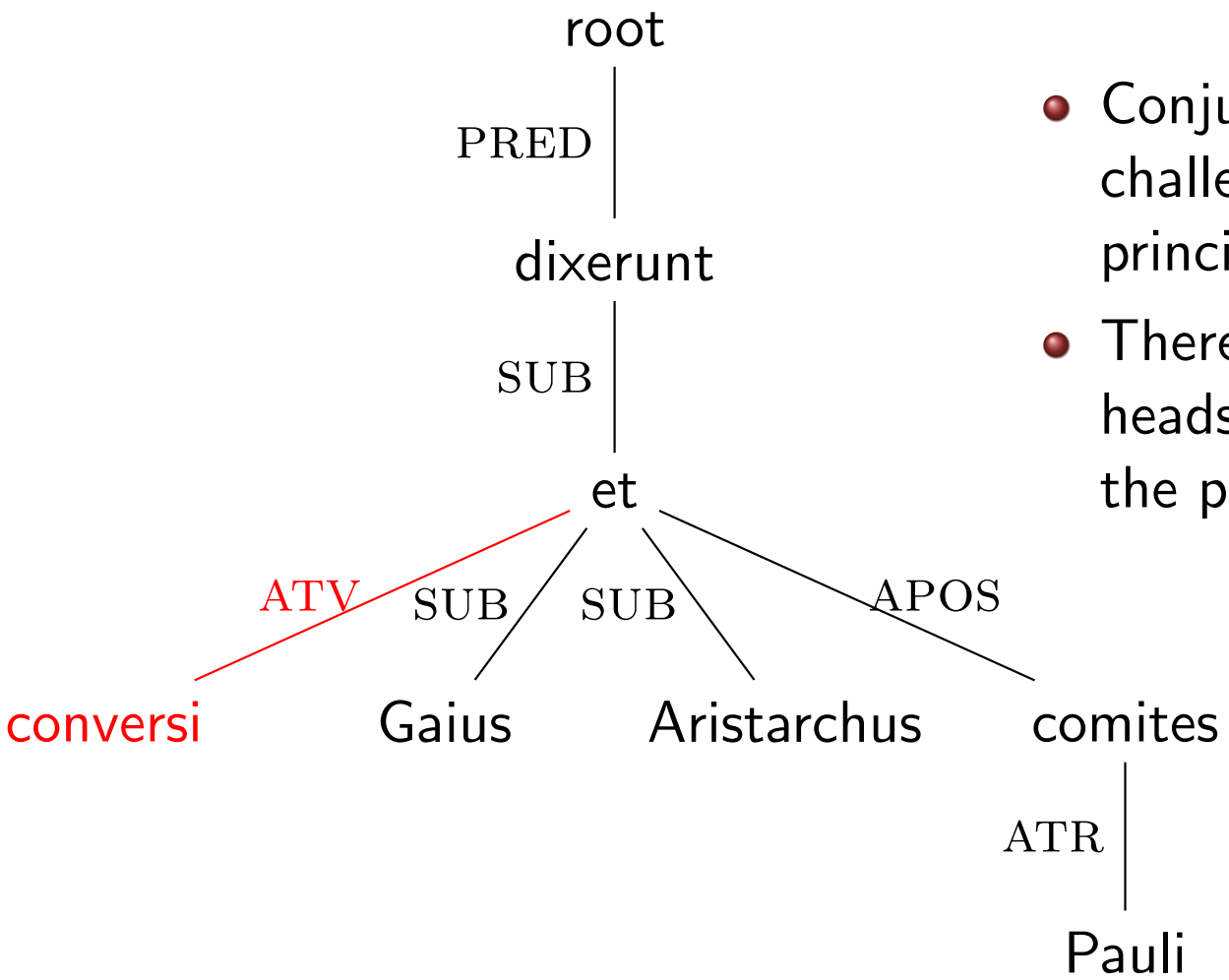
A simple example



- The final result

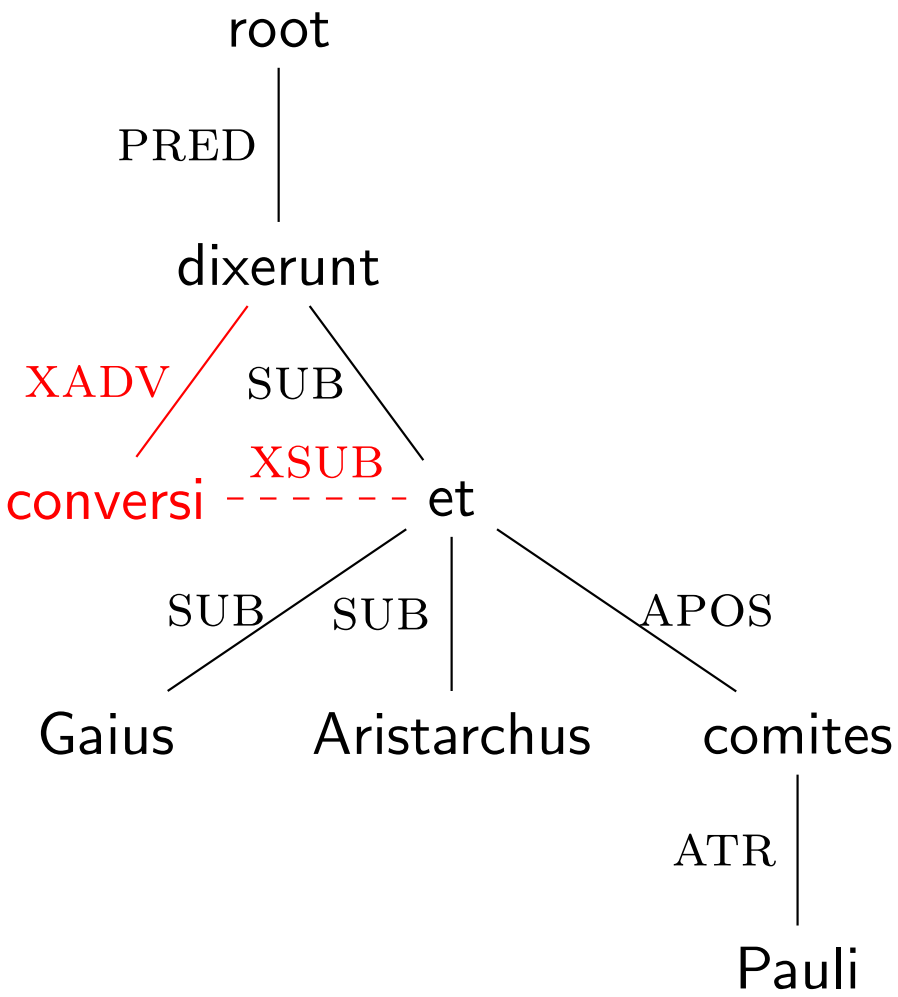
$$\begin{bmatrix} \text{PRED} & \text{'AMARE}\ \langle \text{SUBJ, OBJ} \rangle\text{'} \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{'PUELLA'} \\ \text{CASE} & \text{ACC} \\ \text{GEND} & \text{FEM} \\ \text{ADJ} & \left\{ \begin{bmatrix} \text{PRED} & \text{'PULCHER'} \\ \text{CASE} & \text{ACC} \\ \text{GEND} & \text{FEM} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Structure sharing 1



- Conjunct participles challenge the unique head principle
- There are two candidate heads: the main verb and the participle subject

Structure sharing 2

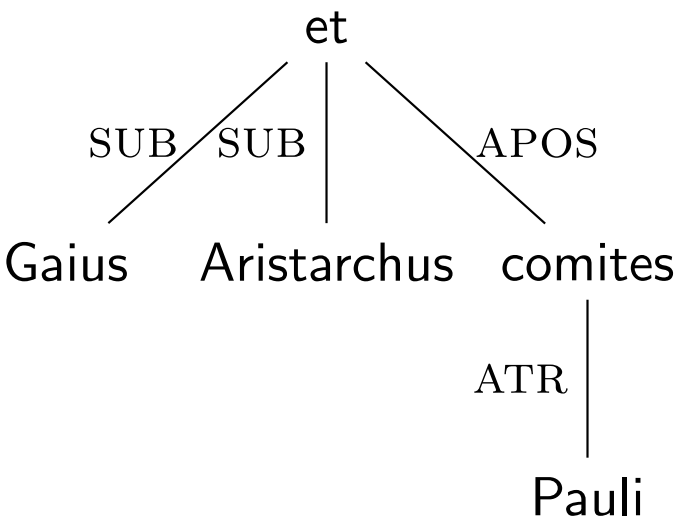


- With secondary edges we can represent both dependencies

F-structure representation

$$\left[\begin{array}{ll} \text{PRED} & \text{'DICO } \langle \text{SUB, OBL, COMP} \rangle \text{'} \\ \text{SUBJ} & \ldots \\ \text{XADJ} & \left\{ \left[\begin{array}{ll} \text{PRED} & \text{'CONVERTOR } \langle \text{SUB} \rangle \text{'} \\ \text{SUBJ} & \ldots \end{array}\right] \right\} \end{array}\right]$$

Features in coordination



$$\left[\begin{array}{ll} \text{NUM} & \text{pl} \\ \left\{\left[\begin{array}{ll} \text{ADJ} & \left\{\left[\begin{array}{ll} \text{PRED} & \text{'COMES}\ \langle \text{OBL} \rangle\text{'} \\ \text{OBL} & \left[\text{PRED}\quad \text{'PAULUS'}\right] \end{array}\right]\right\} \\ \text{PRED} & \text{'GAIUS'} \\ \text{NUM} & \text{SG} \end{array}\right], \left[\begin{array}{ll} \text{ADJ} & \{\text{—}\} \\ \text{PRED} & \text{'ARISTARCHUS'} \\ \text{NUM} & \text{SG} \end{array}\right]\right\} \end{array}\right]$$

- The adjunct is a distributive feature
- Non-distributive features are computed from the set members
- Number, gender and person are such features

Preliminaries

- C-structures and DGs contain very different information

Preliminaries

- C-structures and DGs contain very different information
- Instead of syntactic dependencies, c-structures contain information about
  - category
  - word order
  - word groupings (constituents)

Preliminaries

- C-structures and DGs contain very different information
- Instead of syntactic dependencies, c-structures contain information about
  - category
  - word order
  - word groupings (constituents)
- Of these, only word order is present in a DG (assuming there is a precedence order on terminals)

Preliminaries

- C-structures and DGs contain very different information
- Instead of syntactic dependencies, c-structures contain information about
  - category
  - word order
  - word groupings (constituents)
- Of these, only word order is present in a DG (assuming there is a precedence order on terminals)
- We will see how we can enrich DGs with ‘projections’ that include the other information

Preliminaries

- C-structures and DGs contain very different information
- Instead of syntactic dependencies, c-structures contain information about
  - category
  - word order
  - word groupings (constituents)
- Of these, only word order is present in a DG (assuming there is a precedence order on terminals)
- We will see how we can enrich DGs with 'projections' that include the other information
- The makeup of constituents is a matter of theoretical debate, so we need to introduce theoretical assumptions from LFG

Basic DG

What's in a DG?

A DG is a tuple $\langle \mathcal{W}, r, R_{\mathcal{D}} \rangle$ where

- $\mathcal{W}$ is the set of words totally ordered by $\prec$
- $R_{\mathcal{D}}$ is a set of dependency relations that forms a tree over $\mathcal{W}$ rooted in $r(\in \mathcal{W})$

DG with categories

- The basic point is to note that category constraints are in principle independent of other constraints

DG with categories

- The basic point is to note that category constraints are in principle independent of other constraints
- The classic case is the German Mittelfeld (Bröker 1998)

DG with categories

- The basic point is to note that category constraints are in principle independent of other constraints
- The classic case is the German Mittelfeld (Bröker 1998)
- We can simply extend our model with a class of categories $\mathcal{C}$ and a function $V_{\mathcal{C}} : \mathcal{W} \mapsto \mathcal{C}$

DG with categories

- The basic point is to note that category constraints are in principle independent of other constraints
- The classic case is the German Mittelfeld (Bröker 1998)
- We can simply extend our model with a class of categories $\mathcal{C}$ and a function $V_{\mathcal{C}} : \mathcal{W} \mapsto \mathcal{C}$
- In practice we will use the morphological annotations on the words and map them to a set of theoretically motivated categories

DG with categories

- The basic point is to note that category constraints are in principle independent of other constraints
- The classic case is the German Mittelfeld (Bröker 1998)
- We can simply extend our model with a class of categories $\mathcal{C}$ and a function $V_{\mathcal{C}} : \mathcal{W} \mapsto \mathcal{C}$
- In practice we will use the morphological annotations on the words and map them to a set of theoretically motivated categories
- Notice that if we conceive of $V_{\mathcal{C}}$ as a projection, it is different from LFG projections since it embodies linguistic knowledge (the ϕ function is not similarly restricted)

Order domains (Adapted from Bröker 1998)

Definition

The order domain $\mathcal{D}_w$ of a word w is the largest subset of $\mathcal{W}$ such that

1. $w \in \mathcal{D}_w$

Order domains (Adapted from Bröker 1998)

Definition

The order domain $\mathcal{D}_w$ of a word w is the largest subset of $\mathcal{W}$ such that

1. $w \in \mathcal{D}_w$
2. all words in $\mathcal{D}_w$ are dominated by w

Order domains (Adapted from Bröker 1998)

Definition

The order domain $\mathcal{D}_w$ of a word w is the largest subset of $\mathcal{W}$ such that

1. $w \in \mathcal{D}_w$
2. all words in $\mathcal{D}_w$ are dominated by w
3. $\mathcal{D}_w$ is continuous, i.e. for any two words in $\mathcal{D}_w$, all words in between are also contained in $\mathcal{D}_w$

Order domains (Adapted from Bröker 1998)

Definition

The order domain $\mathcal{D}_w$ of a word w is the largest subset of $\mathcal{W}$ such that

1. $w \in \mathcal{D}_w$
2. all words in $\mathcal{D}_w$ are dominated by w
3. $\mathcal{D}_w$ is continuous, i.e. for any two words in $\mathcal{D}_w$, all words in between are also contained in $\mathcal{D}_w$

- Intuitively, the order domain corresponds to all of the node's dependents that are not 'displaced'

Order domain structures

Order domain structure

The set of order domains of all words $w \in \mathcal{W}$ is a semi-lattice ordered by set inclusion. The join/meet of the semi-lattice is $\mathcal{W}$.

- Every order domain is included in exactly one other order domain, and the order domains are ordered by precedence so the order domain structure is in effect an ordered tree

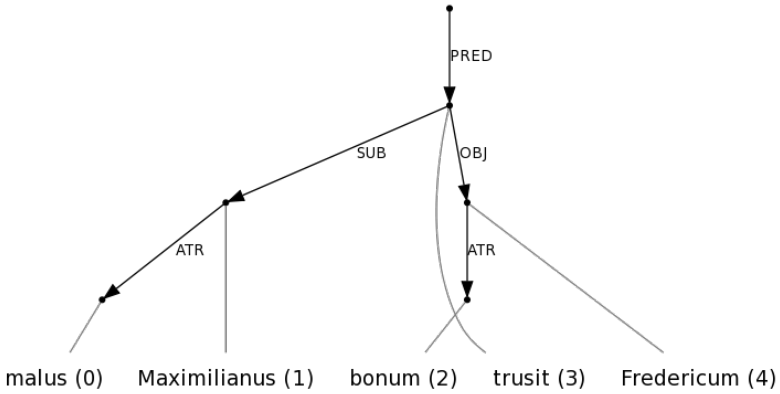
Order domain structures

Order domain structure

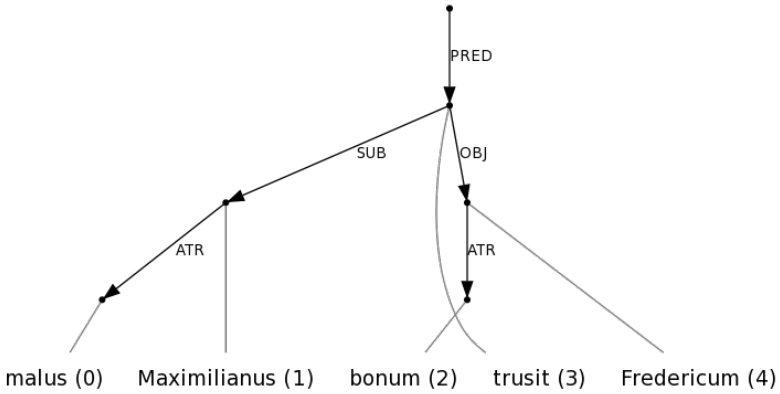
The set of order domains of all words $w \in \mathcal{W}$ is a semi-lattice ordered by set inclusion. The join/meet of the semi-lattice is $\mathcal{W}$.

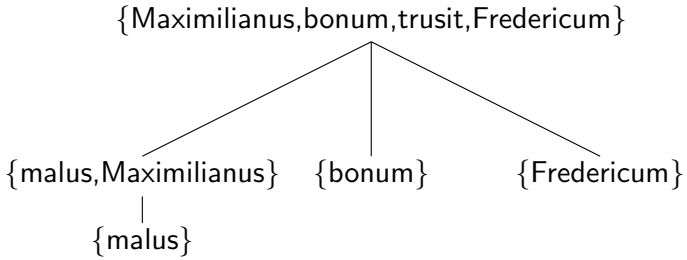
- Every order domain is included in exactly one other order domain, and the order domains are ordered by precedence so the order domain structure is in effect an ordered tree
- Similar to those generated by CFGs but without the categorial information

Example

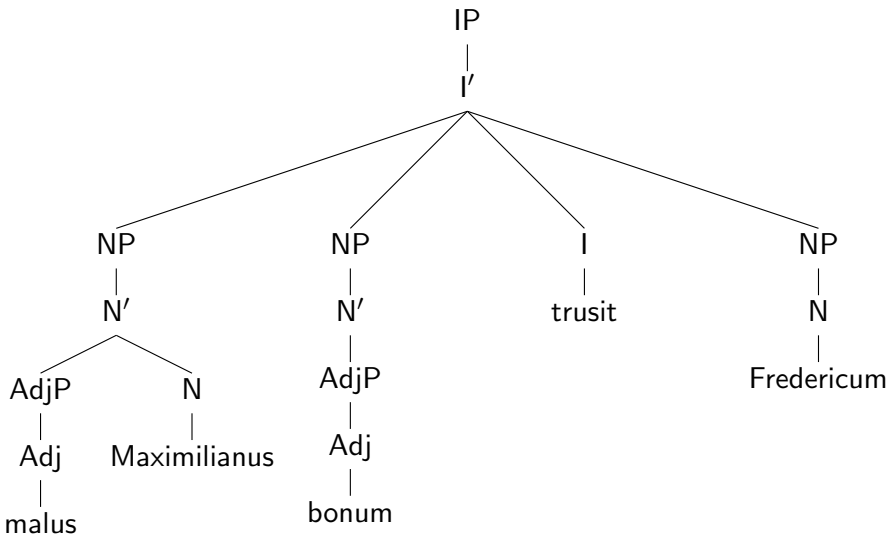

PRED
SUB
OBJ
ATR
ATR
malus (0)
Maximilianus (1)
bonum (2)
trusit (3)
Fredericum (4)

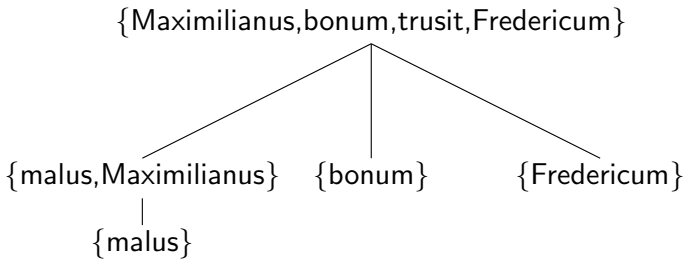
Example


PRED
SUB
OBJ
ATR
ATR
malus (0)
Maximilianus (1)
bonum (2)
trusit (3)
Fredericum (4)


{Maximilianus,bonum,trusit,Fredericum}
{malus,Maximilianus}
{bonum}
{Fredericum}
{malus}

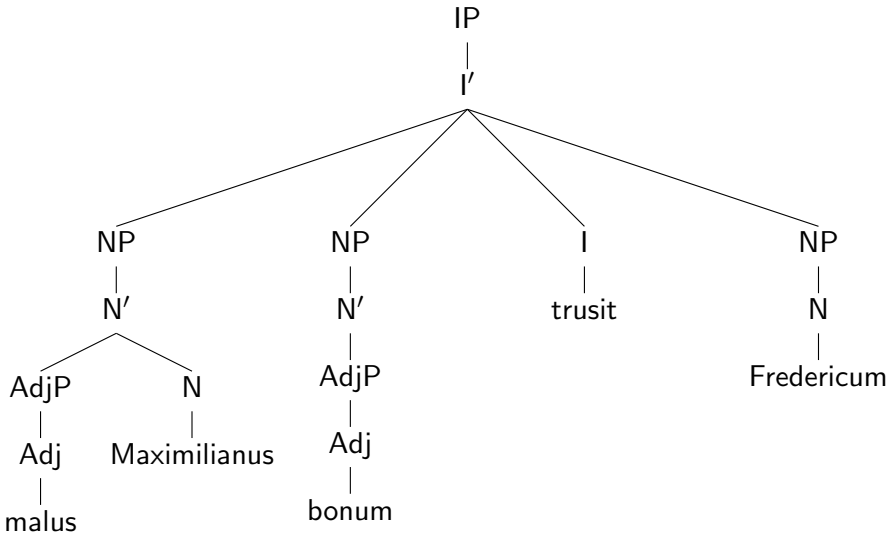
Example

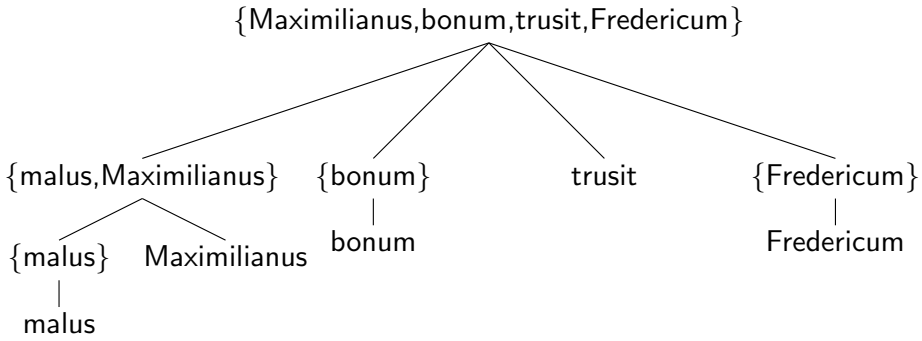




- Each Bröker node corresponds to a X″ - X′ - X spine

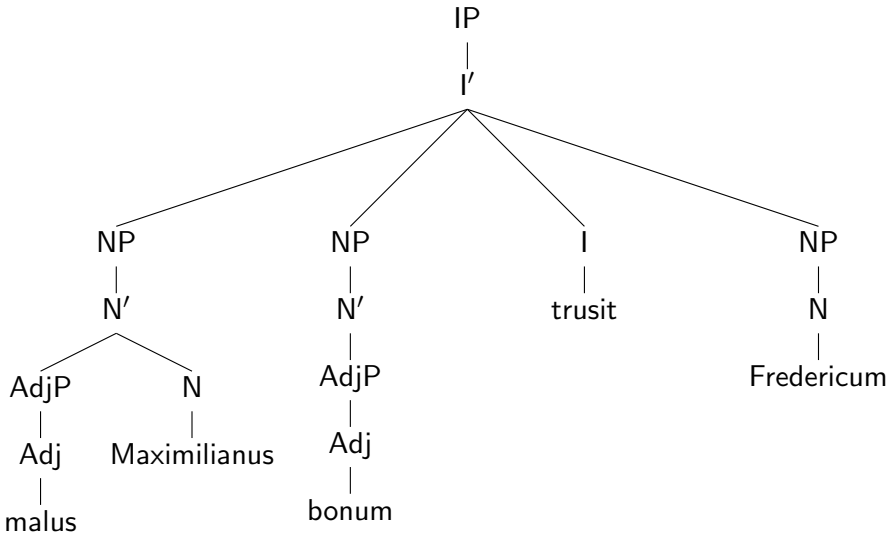
Example

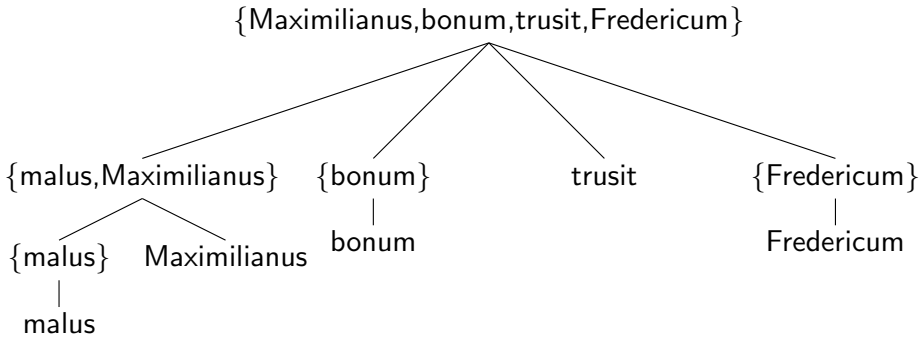




- Each Bröker node corresponds to a X″ - X′ - X spine
- We can add explicit heads (each w is the head of $\mathcal{D}_w$)

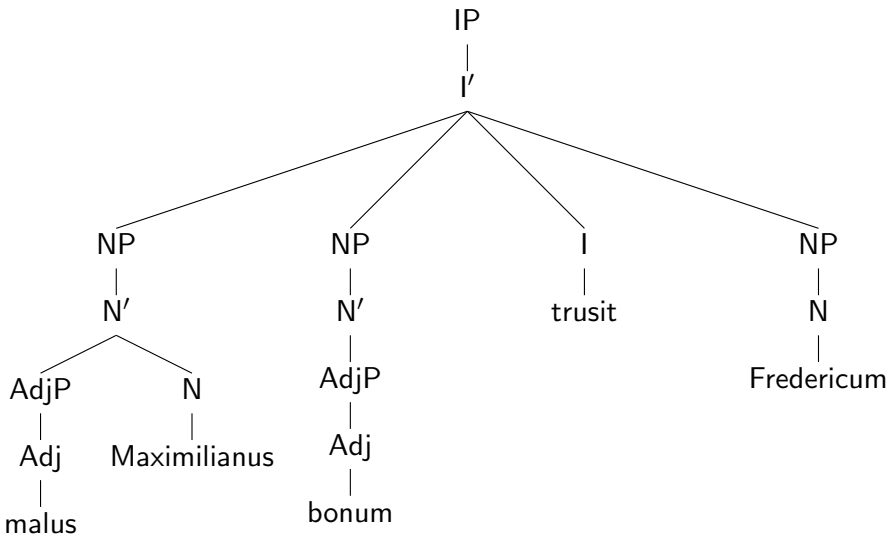
Example

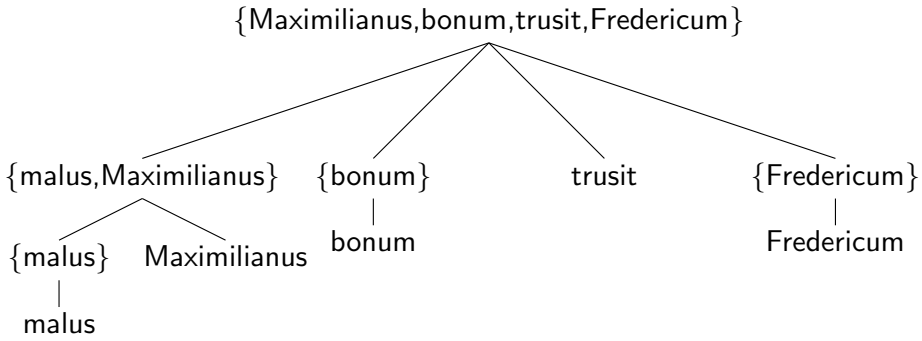




- Each Bröker node corresponds to a X″ - X′ - X spine
- We can add explicit heads (each w is the head of $\mathcal{D}_w$)
- Probably as close as we can come in a pure projection from the DG

Example





- Each Bröker node corresponds to a X″ - X′ - X spine
- We can add explicit heads (each w is the head of $\mathcal{D}_w$)
- Probably as close as we can come in a pure projection from the DG
- What we are lacking is a theory of the internal structure of phrases

Internal structure of phrases

Questions (from Xia 2001)

1. for a category X, what kind of projections can X have?
2. if a category Y depends on a category Y in a dependency structure, how far should Y project before it attaches to Xs projection?
3. if a category Y depends on a category X in a dependency structure, to what position on X's projection chain should Y's projection attach?

Internal structure of phrases

Answers

1. all categories X project two levels X′ and XP.
2. a dependent Y always projects to Y′ then YP and the YP attaches to the head's projection
3. dependents are divided into three types using a set of handwritten rules: specifiers, modifiers and arguments. Specifiers are made sisters of X′ and arguments are made daughters of X. Modifiers Chomsky-adjoin to either X′ or XP depending on whether they are restrictive, as indicated by the dependency edge label (ATR or APOS).

An algorithm

$\mathcal{L} = \{\}$
function CreateProjection(n)
  $\mathcal{D} = \{\}$
  for all *d*: daughters of *n* **do**
    put CreateProjection(d) in $\mathcal{D}$
  end for
  for all $d \in \mathcal{D} \cup \mathcal{L}$ **do**
    if *d* is in n's order domain **then**
      put/leave *d*' in $\mathcal{D}$
    else
      put/leave *d* in $\mathcal{L}$
    end if
  end for
  make the elements in $\mathcal{D}$ daughters of *n*
end function

Adding linguistic knowledge

- This algorithm gives us the Bröker trees

Adding linguistic knowledge

- This algorithm gives us the Bröker trees
- We can enrich these with linguistic knowledge

Adding linguistic knowledge

- This algorithm gives us the Bröker trees
- We can enrich these with linguistic knowledge
- We will use our X′ assumptions, the category mapping and handwritten phrase structure rules

A sample rule

```
N:
    :phrase_adjuncts:
        - NP
        - AdjP
    :specifier:
        - DP
    :bar_adjuncts:
        - AdjP
        - NP
    :complements:
        - NP
```

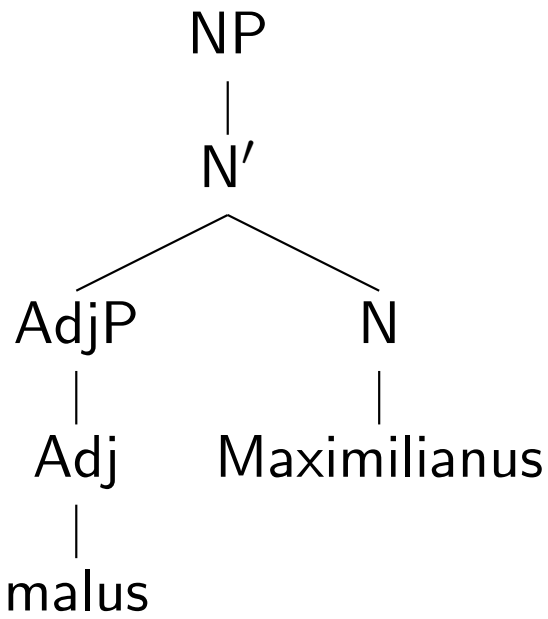
Adding linguistic knowledge

- This algorithm gives us the Bröker trees
- We can enrich these with linguistic knowledge
- We will use our X′ assumptions, the category mapping and handwritten phrase structure rules
- We can recursively embed loose nodes under headless structures to achieve the LFG analysis of non-projectivity

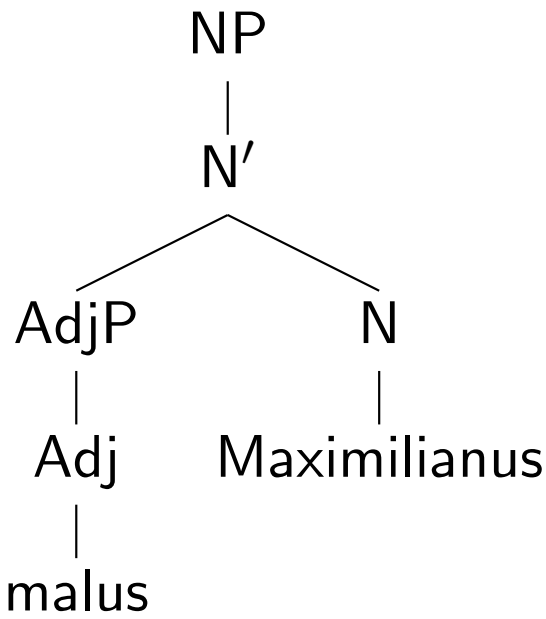
Where to add linguistics

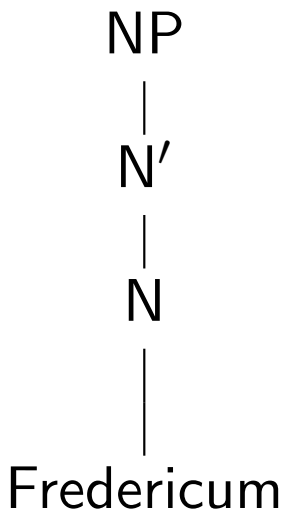
$\mathcal{L} = \{\}$
function CreateProjection(n)
  $\mathcal{D} = \{\}$
  for all *d*: daughters of *n* **do**
    put CreateProjection(d) in $\mathcal{D}$
  end for
  for all $d \in \mathcal{D} \cup \mathcal{L}$ **do**
    if *d* is in n's order domain **then**
      put/leave *d*' in $\mathcal{D}$
    else
      put/leave *d* in $\mathcal{L}$
    end if
  end for
  make the elements in $\mathcal{D}$ daughters of *n*
end function

$\mathcal{D}$ $\mathcal{L}$

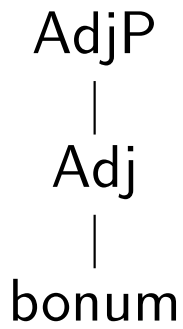


$\mathcal{D}$

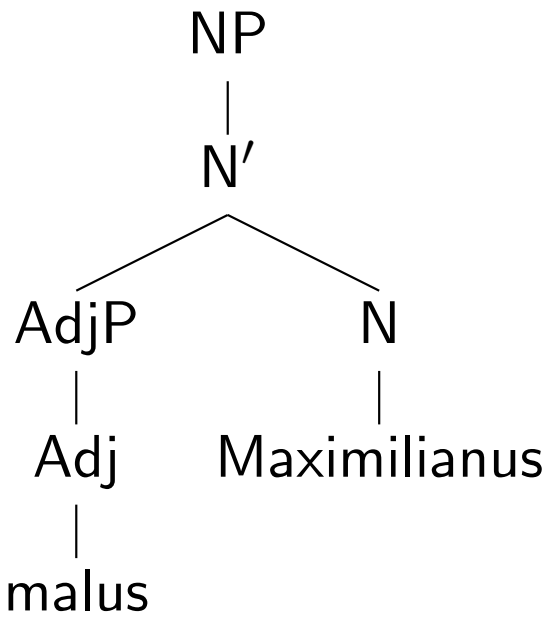


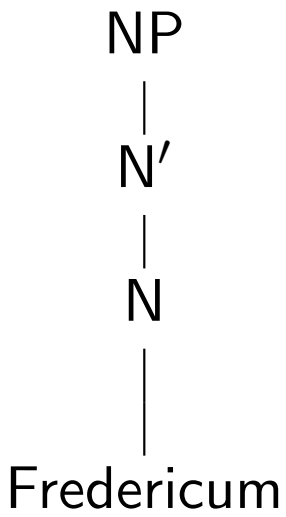


$\mathcal{L}$

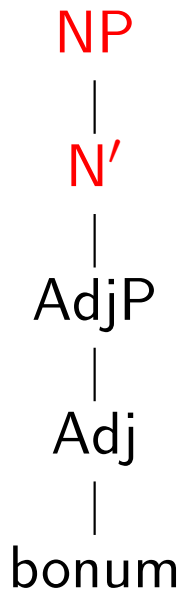


$\mathcal{D}$

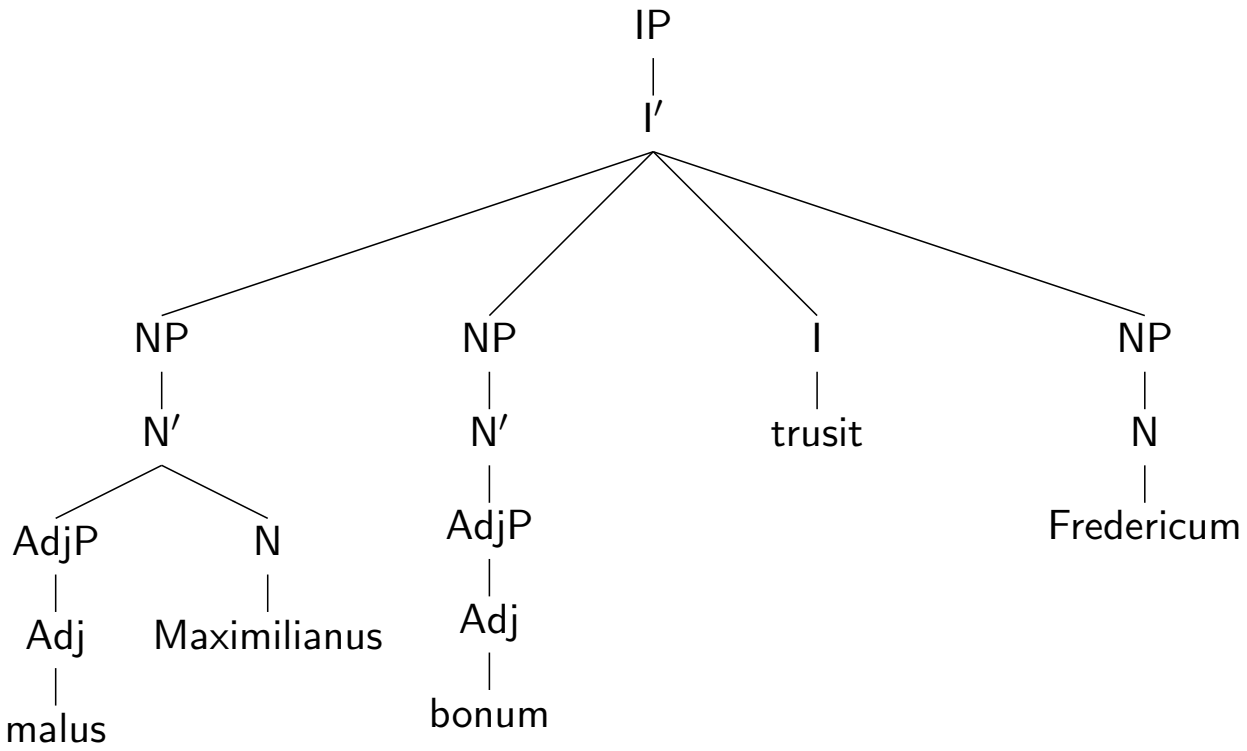




$\mathcal{L}$



The result



▸ Hyperbaton ▸ WH-movement

Summary

- We have seen that the PROIEL corpus is a small but deeply annotated corpus
  - Morphology
  - Syntax
  - Information structure
  - Discourse (experimental, not shown)

Summary

- We have seen that the PROIEL corpus is a small but deeply annotated corpus
  - Morphology
  - Syntax
  - Information structure
  - Discourse (experimental, not shown)
- The syntax is as theory-neutral as possible

Summary

- We have seen that the PROIEL corpus is a small but deeply annotated corpus
  - Morphology
  - Syntax
  - Information structure
  - Discourse (experimental, not shown)
- The syntax is as theory-neutral as possible
- But conversion is possible and an interesting for hypothesis testing

Summary

- We have seen that the PROIEL corpus is a small but deeply annotated corpus
  - Morphology
  - Syntax
  - Information structure
  - Discourse (experimental, not shown)
- The syntax is as theory-neutral as possible
- But conversion is possible and an interesting for hypothesis testing
- The output could be used as a test suite for a implementing an LFG grammar

Summary

- We have seen that the PROIEL corpus is a small but deeply annotated corpus
  - Morphology
  - Syntax
  - Information structure
  - Discourse (experimental, not shown)
- The syntax is as theory-neutral as possible
- But conversion is possible and an interesting for hypothesis testing
- The output could be used as a test suite for a implementing an LFG grammar
- It can also make the data more widely available to researchers in other frameworks

Outlook

- The New Testament text is available for many low-resources languages

Outlook

- The New Testament text is available for many low-resources languages
- The fine-grained reference system (book, chapter, verse) makes alignment feasible

Outlook

- The New Testament text is available for many low-resources languages
- The fine-grained reference system (book, chapter, verse) makes alignment feasible
- We will experiment with annotation transfer
  - Cooperation with the Linguistic Data Consortium at Penn: alignment, comparison, annotation transfer with phrase structure-based NT corpora

Outlook

- The New Testament text is available for many low-resources languages
- The fine-grained reference system (book, chapter, verse) makes alignment feasible
- We will experiment with annotation transfer
  - Cooperation with the Linguistic Data Consortium at Penn: alignment, comparison, annotation transfer with phrase structure-based NT corpora
  - Cooperation with Iceland and Språkbanken in Gothenburg: alignment and annotation transfer between annotated and unannotated, Nordic bible texts (Old Swedish, Icelandic, possibly Old Finnish)

Availability

- The corpus is available for everyone to use.
- We publish XML files with raw data as well.
- All our data is released under a Creative Commons license.
- Visit `http://www.hf.uio.no/ifikk/proiel/` for details.