Outline of today's lecture

Beyond simple CFGs

Feature structures (informally)

Encoding agreement

Parsing with feature structures

Feature stuctures more formally

Encoding subcategorisation

Interface to morphology

Long-distance dependencies

1. which problem did you say you don't understand?
2. who do you think Kim asked Sandy to hit?
3. which kids did you say were making all that noise?

'gaps' (underscores below)

1. which problem did you say you don't understand _?
2. who do you think Kim asked Sandy to hit _?
3. which kids did you say _ were making all that noise?

In 3, the verb *were* shows plural agreement.

* what kid did you say _ were making all that noise?

- Informally: need a 'gap' slot which is to be filled by something that itself has features.

Context-free grammar and language phenomena

- CFGs can encode long-distance dependencies
- Language phenomena that CFGs cannot model (without a bound) are unusual — probably none in English.
- BUT: CFG modelling for English or another NL could be trillions of rules
- Enriched formalisms: CFG equivalent or greater power

Constraint-based grammar (feature structures)

Providing a more adequate treatment of syntax than simple CFGs by replacing the atomic categories by more complex data structures.

- Feature structure formalisms give good linguistic accounts for many languages
- Reasonably computationally tractable
- Bidirectional (parse and generate)
- Used in LFG and HPSG formalisms

Can also think of CFGs as constraints on trees.

Expanded CFG (from last time)

```
S -> NP-sg-nom VP-sg          NP-sg-nom -> he
S -> NP-pl-nom VP-pl          NP-sg-acc -> him
VP-sg -> V-sg NP-sg-acc       NP-sg-nom -> fish
VP-sg -> V-sg NP-pl-acc       NP-pl-nom -> fish
VP-pl -> V-pl NP-sg-acc       NP-sg-acc -> fish
VP-pl -> V-pl NP-pl-acc       NP-pl-acc -> fish
```

Intuitive solution for case and agreement

- Separate slots (features) for CASE and AGR
- Slot values for CASE may be **nom** (e.g., *they*), **acc** (e.g., *them*) or unspecified (i.e., don’t care)
- Slot values for AGR may be **sg**, **pl** or unspecified
- Subjects have the same value for AGR as their verbs
- Subjects have CASE **nom**, objects have CASE **acc**

can (n) $\begin{bmatrix} \text{CASE} & [\,] \\ \text{AGR} & \textbf{sg} \end{bmatrix}$ fish (n) $\begin{bmatrix} \text{CASE} & [\,] \\ \text{AGR} & [\,] \end{bmatrix}$

she $\begin{bmatrix} \text{CASE} & \textbf{nom} \\ \text{AGR} & \textbf{sg} \end{bmatrix}$ them $\begin{bmatrix} \text{CASE} & \textbf{acc} \\ \text{AGR} & \textbf{pl} \end{bmatrix}$

Feature structures

$$\begin{bmatrix} \text{CASE} & [\,] \\ \text{AGR} & \textbf{sg} \end{bmatrix}$$

1. Features like AGR with simple values: atomic-valued
2. Unspecified values possible on features: compatible with any value.
3. Values for features for subcat and gap themselves have features: complex-valued
4. path: a sequence of features
5. Method of specifying two paths are the same: reentrancy
6. Unification: combining two feature structures, retaining all information from each, or fail if information is incompatible.

Simple unification examples

1. $$\begin{bmatrix} \text{CASE} & [\,] \\ \text{AGR} & \textbf{sg} \end{bmatrix} \sqcap \begin{bmatrix} \text{CASE} & \textbf{nom} \\ \text{AGR} & [\,] \end{bmatrix} = \begin{bmatrix} \text{CASE} & \textbf{nom} \\ \text{AGR} & \textbf{sg} \end{bmatrix}$$

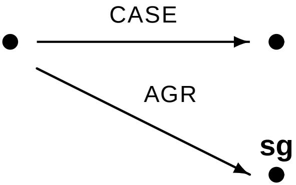
2. $$\begin{bmatrix} \text{CASE} & [\,] \\ \text{AGR} & \textbf{sg} \end{bmatrix} \sqcap \begin{bmatrix} \text{AGR} & [\,] \end{bmatrix} = \begin{bmatrix} \text{CASE} & [\,] \\ \text{AGR} & \textbf{sg} \end{bmatrix}$$

3. $$\begin{bmatrix} \text{CASE} & [\,] \\ \text{AGR} & \textbf{sg} \end{bmatrix} \sqcap \begin{bmatrix} \text{CASE} & \textbf{nom} \\ \text{AGR} & \textbf{pl} \end{bmatrix} = \text{fail}$$

Feature structures, continued

- Feature structures are singly-rooted directed acyclic graphs, with arcs labelled by features and terminal nodes associated with values.

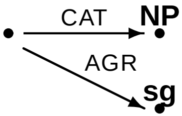
$$\begin{bmatrix} \text{CASE} & [\,] \\ \text{AGR} & \textbf{sg} \end{bmatrix}$$



- In grammars, rules relate FSs — i.e. lexical entries and phrases are represented as FSs
- Rule application by unification

Graphs and AVMs

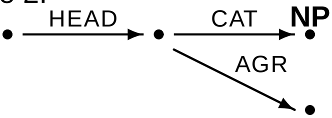
Example 1:



$$\begin{bmatrix} \text{CAT} & \textbf{NP} \\ \text{AGR} & \textbf{sg} \end{bmatrix}$$

Here, CAT and AGR are atomic-valued features. **NP** and **sg** are values.

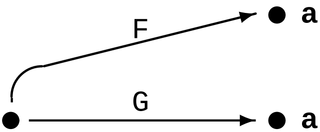
Example 2:



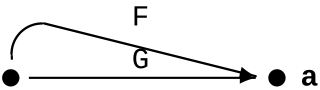
$$\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \textbf{NP} \\ \text{AGR} & [\,] \end{bmatrix} \end{bmatrix}$$

HEAD is complex-valued, AGR is unspecified.

Reentrancy



$$\begin{bmatrix} \text{F} & \mathbf{a} \\ \text{G} & \mathbf{a} \end{bmatrix}$$



$$\begin{bmatrix} \text{F} & \boxed{0}\ \mathbf{a} \\ \text{G} & \boxed{0} \end{bmatrix}$$

Reentrancy indicated by boxed integer in AVM diagram: indicates path goes to the same node.

CFG with agreement

```
S -> NP-sg VP-sg
S -> NP-pl VP-pl
VP-sg -> V-sg NP-sg
VP-sg -> V-sg NP-pl
VP-pl -> V-pl NP-sg
VP-pl -> V-pl NP-pl
V-pl -> like
V-sg -> likes
NP-sg -> it
NP-pl -> they
NP-sg -> fish
NP-pl -> fish
```

FS grammar fragment encoding agreement

subj-verb rule $\begin{bmatrix} \text{CAT} & \mathbf{S} \\ \text{AGR} & \boxed{1} \end{bmatrix} \rightarrow \begin{bmatrix} \text{CAT} & \mathbf{NP} \\ \text{AGR} & \boxed{1} \end{bmatrix}, \begin{bmatrix} \text{CAT} & \mathbf{VP} \\ \text{AGR} & \boxed{1} \end{bmatrix}$

verb-obj rule $\begin{bmatrix} \text{CAT} & \mathbf{VP} \\ \text{AGR} & \boxed{1} \end{bmatrix} \rightarrow \begin{bmatrix} \text{CAT} & \mathbf{V} \\ \text{AGR} & \boxed{1} \end{bmatrix}, \begin{bmatrix} \text{CAT} & \mathbf{NP} \\ \text{AGR} & [\,] \end{bmatrix}$

Root structure: $\begin{bmatrix} \text{CAT} & \mathbf{S} \end{bmatrix}$

they $\begin{bmatrix} \text{CAT} & \mathbf{NP} \\ \text{AGR} & \mathbf{pl} \end{bmatrix}$ it $\begin{bmatrix} \text{CAT} & \mathbf{NP} \\ \text{AGR} & \mathbf{sg} \end{bmatrix}$ likes $\begin{bmatrix} \text{CAT} & \mathbf{V} \\ \text{AGR} & \mathbf{sg} \end{bmatrix}$

fish $\begin{bmatrix} \text{CAT} & \mathbf{NP} \\ \text{AGR} & [\,] \end{bmatrix}$ like $\begin{bmatrix} \text{CAT} & \mathbf{V} \\ \text{AGR} & \mathbf{pl} \end{bmatrix}$

Parsing ‘they like it’

- The lexical structures for *like* and *it* are unified with the corresponding structures on the right hand side of the verb-obj rule (unifications succeed).
- The structure corresponding to the mother of the rule is then:

$$\begin{bmatrix} \text{CAT} & \textbf{VP} \\ \text{AGR} & \textbf{pl} \end{bmatrix}$$

- This unifies with the rightmost daughter position of the subj-verb rule.
- The structure for *they* is unified with the leftmost daughter.
- The result unifies with root structure.

Rules as FSs

But what does the coindexation of parts of the rule mean? Treat rule as a FS: e.g., rule features MOTHER, DTR1, DTR2 ... DTRN.

informally: $\begin{bmatrix} \text{CAT} & \textbf{VP} \\ \text{AGR} & \boxed{1} \end{bmatrix} \rightarrow \begin{bmatrix} \text{CAT} & \textbf{V} \\ \text{AGR} & \boxed{1} \end{bmatrix}, \begin{bmatrix} \text{CAT} & \textbf{NP} \\ \text{AGR} & [\,] \end{bmatrix}$

actually: $\begin{bmatrix} \text{MOTHER} & \begin{bmatrix} \text{CAT} & \textbf{VP} \\ \text{AGR} & \boxed{1} \end{bmatrix} \\ \text{DTR1} & \begin{bmatrix} \text{CAT} & \textbf{V} \\ \text{AGR} & \boxed{1} \end{bmatrix} \\ \text{DTR2} & \begin{bmatrix} \text{CAT} & \textbf{NP} \\ \text{AGR} & [\,] \end{bmatrix} \end{bmatrix}$

Verb-obj rule application

Feature structure for *like* unified with the value of DTR1:

$$\begin{bmatrix} \text{MOTHER} & \begin{bmatrix} \text{CAT} & \textbf{VP} \\ \text{AGR} & \boxed{1}\ \textbf{pl} \end{bmatrix} \\ \text{DTR1} & \begin{bmatrix} \text{CAT} & \textbf{V} \\ \text{AGR} & \boxed{1} \end{bmatrix} \\ \text{DTR2} & \begin{bmatrix} \text{CAT} & \textbf{NP} \\ \text{AGR} & [\,] \end{bmatrix} \end{bmatrix}$$

Feature structure for *it* unified with the value for DTR2:

$$\begin{bmatrix} \text{MOTHER} & \begin{bmatrix} \text{CAT} & \textbf{VP} \\ \text{AGR} & \boxed{1}\ \textbf{pl} \end{bmatrix} \\ \text{DTR1} & \begin{bmatrix} \text{CAT} & \textbf{V} \\ \text{AGR} & \boxed{1} \end{bmatrix} \\ \text{DTR2} & \begin{bmatrix} \text{CAT} & \textbf{NP} \\ \text{AGR} & \textbf{sg} \end{bmatrix} \end{bmatrix}$$

Subject-verb rule application 1

MOTHER value from the verb-object rule acts as the DTR2 of the subject-verb rule:

$\begin{bmatrix} \text{CAT} & \textbf{VP} \\ \text{AGR} & \textbf{pl} \end{bmatrix}$ unified with the DTR2 of:
$$\begin{bmatrix} \text{MOTHER} & \begin{bmatrix} \text{CAT} & \textbf{S} \\ \text{AGR} & \boxed{1} \end{bmatrix} \\ \text{DTR1} & \begin{bmatrix} \text{CAT} & \textbf{NP} \\ \text{AGR} & \boxed{1} \end{bmatrix} \\ \text{DTR2} & \begin{bmatrix} \text{CAT} & \textbf{VP} \\ \text{AGR} & \boxed{1} \end{bmatrix} \end{bmatrix}$$

Gives:

$$\begin{bmatrix} \text{MOTHER} & \begin{bmatrix} \text{CAT} & \textbf{S} \\ \text{AGR} & \boxed{1}\ \textbf{pl} \end{bmatrix} \\ \text{DTR1} & \begin{bmatrix} \text{CAT} & \textbf{NP} \\ \text{AGR} & \boxed{1} \end{bmatrix} \\ \text{DTR2} & \begin{bmatrix} \text{CAT} & \textbf{VP} \\ \text{AGR} & \boxed{1} \end{bmatrix} \end{bmatrix}$$

Subject rule application 2

FS for *they*: $\begin{bmatrix} \text{CAT} & \textbf{NP} \\ \text{AGR} & \textbf{pl} \end{bmatrix}$

Unification of this with the value of DTR1 succeeds (but adds no new information):

$$\begin{bmatrix} \text{MOTHER} & \begin{bmatrix} \text{CAT} & \textbf{S} \\ \text{AGR} & \boxed{1}\ \textbf{pl} \end{bmatrix} \\ \text{DTR1} & \begin{bmatrix} \text{CAT} & \textbf{NP} \\ \text{AGR} & \boxed{1} \end{bmatrix} \\ \text{DTR2} & \begin{bmatrix} \text{CAT} & \textbf{VP} \\ \text{AGR} & \boxed{1} \end{bmatrix} \end{bmatrix}$$

Final structure unifies with the root structure: $\begin{bmatrix} \text{CAT} & \textbf{S} \end{bmatrix}$

Properties of FSs

Connectedness and unique root A FS must have a unique root node: apart from the root node, all nodes have one or more parent nodes.

Unique features Any node may have zero or more arcs leading out of it, but the label on each (that is, the feature) must be unique.

No cycles No node may have an arc that points back to the root node or to a node that intervenes between it and the root node.

Values A node which does not have any arcs leading out of it may have an associated atomic value.

Finiteness A FS must have a finite number of nodes.

Subsumption

Feature structures are ordered by information content — FS1 *subsumes* FS2 if FS2 carries extra information.
FS1 subsumes FS2 if and only if the following conditions hold:

Path values For every path P in FS1 there is a path P in FS2. If P has a value t in FS1, then P also has value t in FS2.

Path equivalences Every pair of paths P and Q which are reentrant in FS1 (i.e., which lead to the same node in the graph) are also reentrant in FS2.

Unification
The unification of two FSs FS1 and FS2 is the most general FS which is subsumed by both FS1 and FS2, if it exists.

Grammar with subcategorisation

Verb-obj rule: $\begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{OBJ} & \textbf{filled} \\ \text{SUBJ} & \boxed{3} \end{bmatrix} \rightarrow \begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{OBJ} & \boxed{2} \\ \text{SUBJ} & \boxed{3} \end{bmatrix}, \boxed{2} \begin{bmatrix} \text{OBJ} & \textbf{filled} \end{bmatrix}$

can (transitive verb): $\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \textbf{verb} \\ \text{AGR} & \textbf{pl} \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \textbf{noun} \end{bmatrix} \\ \text{OBJ} & \textbf{filled} \end{bmatrix} \\ \text{SUBJ} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \textbf{noun} \end{bmatrix} \end{bmatrix} \end{bmatrix}$

Grammar with subcategorisation (abbrev for slides)

Verb-obj rule:

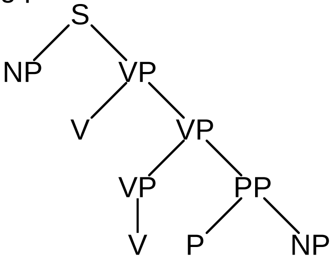
$$\begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \boxed{3} \end{bmatrix} \rightarrow \begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{OBJ} & \boxed{2} \\ \text{SUBJ} & \boxed{3} \end{bmatrix}, \; \boxed{2} \begin{bmatrix} \text{OBJ} & \textbf{fld} \end{bmatrix}$$

can (transitive verb):

$$\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \textbf{v} \\ \text{AGR} & \textbf{pl} \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \textbf{n} \end{bmatrix} \\ \text{OBJ} & \textbf{fld} \end{bmatrix} \\ \text{SUBJ} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \textbf{n} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

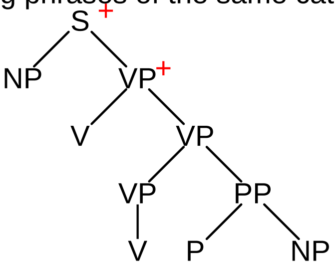
Concepts for subcategorisation

- HEAD: information shared between a lexical entry and the dominating phrases of the same category



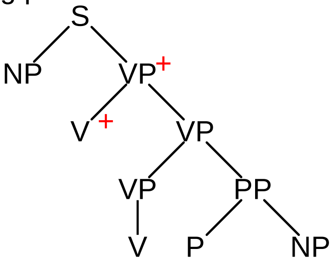
Concepts for subcategorisation

- HEAD: information shared between a lexical entry and the dominating phrases of the same category



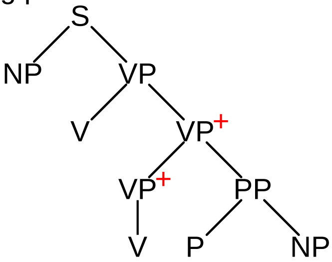
Concepts for subcategorisation

- HEAD: information shared between a lexical entry and the dominating phrases of the same category



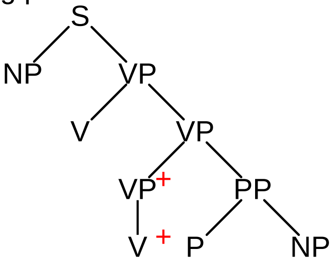
Concepts for subcategorisation

- HEAD: information shared between a lexical entry and the dominating phrases of the same category



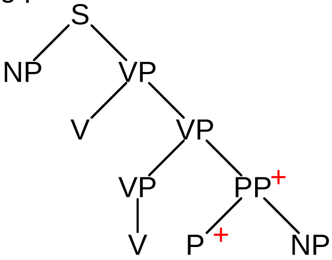
Concepts for subcategorisation

- HEAD: information shared between a lexical entry and the dominating phrases of the same category



Concepts for subcategorisation

- HEAD: information shared between a lexical entry and the dominating phrases of the same category



Concepts for subcategorisation

- HEAD: information shared between a lexical entry and the dominating phrases of the same category
- SUBJ:
  The subject-verb rule unifies the first daughter of the rule with the SUBJ value of the second. (‘the first dtr fills the SUBJ slot of the second dtr in the rule’)

Concepts for subcategorisation

- HEAD: information shared between a lexical entry and the dominating phrases of the same category
- SUBJ:
  The subject-verb rule unifies the first daughter of the rule with the SUBJ value of the second. (‘the first dtr fills the SUBJ slot of the second dtr in the rule’)
- OBJ:
  The verb-object rule unifies the second dtr with the OBJ value of the first. (‘the second dtr fills the OBJ slot of the first dtr in the rule’)

Example rule application: *they fish* 1

Lexical entry for fish:
$$\begin{bmatrix} \text{HEAD} & \begin{bmatrix}\text{CAT} & \textbf{v}\\ \text{AGR} & \textbf{pl}\end{bmatrix} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \begin{bmatrix}\text{HEAD} & \begin{bmatrix}\text{CAT} & \textbf{n}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

subject-verb rule:

$$\begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \textbf{fld} \end{bmatrix} \rightarrow \boxed{2}\begin{bmatrix} \text{HEAD} & \begin{bmatrix}\text{AGR} & \boxed{3}\end{bmatrix} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \textbf{fld} \end{bmatrix}, \begin{bmatrix} \text{HEAD} & \boxed{1}\begin{bmatrix}\text{AGR} & \boxed{3}\end{bmatrix} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \boxed{2} \end{bmatrix}$$

unification with second dtr position gives:

$$\begin{bmatrix} \text{HEAD} & \boxed{1}\begin{bmatrix}\text{CAT} & \textbf{v}\\ \text{AGR} & \boxed{3}\ \textbf{pl}\end{bmatrix} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \textbf{fld} \end{bmatrix} \rightarrow \boxed{2}\begin{bmatrix} \text{HEAD} & \begin{bmatrix}\text{CAT} & \textbf{n}\\ \text{AGR} & \boxed{3}\end{bmatrix} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \textbf{fld} \end{bmatrix}, \begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \boxed{2} \end{bmatrix}$$

Lexical entry for *they*: $\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \textbf{n} \\ \text{AGR} & \textbf{pl} \end{bmatrix} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \textbf{fld} \end{bmatrix}$

unify this with first dtr position:

$$\begin{bmatrix} \text{HEAD} & \boxed{1} \begin{bmatrix} \text{CAT} & \textbf{v} \\ \text{AGR} & \boxed{3}\ \textbf{pl} \end{bmatrix} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \textbf{fld} \end{bmatrix} \rightarrow \boxed{2} \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \textbf{n} \\ \text{AGR} & \boxed{3} \end{bmatrix} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \textbf{fld} \end{bmatrix}, \begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \boxed{2} \end{bmatrix}$$

Root is: $\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \textbf{v} \end{bmatrix} \\ \text{OBJ} & \textbf{fld} \\ \text{SUBJ} & \textbf{fld} \end{bmatrix}$

Mother structure unifies with root, so valid.

Parsing with feature structure grammars

- Naive algorithm: standard chart parser with modified rule application
- Rule application:
  1. copy rule
  2. copy daughters (lexical entries or FSs associated with edges)
  3. unify rule and daughters
  4. if successful, add new edge to chart with rule FS as category
- Efficient algorithms reduce copying.
- Packing involves subsumption.

Templates

Capture generalizations in the lexicon:

fish INTRANS_VERB
sleep INTRANS_VERB
snore INTRANS_VERB

$$\text{INTRANS\_VERB} \qquad \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \mathbf{v} \\ \text{AGR} & \mathbf{pl} \end{bmatrix} \\ \text{OBJ} & \mathbf{fld} \\ \text{SUBJ} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \mathbf{n} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Interface to morphology: inflectional affixes as FSs

s `PLURAL_NOUN` $\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{CAT} & \mathbf{n} \\ \text{AGR} & \mathbf{pl} \end{bmatrix} \end{bmatrix}$

Syntactic effects of affix: stem FS is unified with affix FS.

Unification failure would occur with verbs etc, so we get filtering (lecture 2).

Summary

- Feature structure formalisms can be used to capture linguistic intuitions more directly than CFGs.
- Phenomena: agreement and subcategorisation (but tip of an enormous iceberg).
- Next lecture: Compositional semantics: the construction of meaning based on syntax.