L95: Natural Language Syntax and Parsing
8) Unification-based Grammars and Parsing

Paula Buttery

Dept of Computer Science & Technology, University of Cambridge

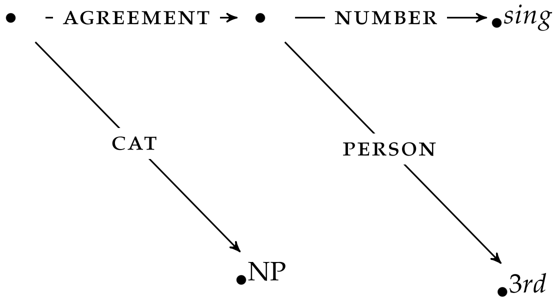
Reminder...

Last time we looked at lexicalisation and features to help us with:

- modelling structural dependency across the tree as a whole
  - e.g. correctly modelling *NP* expansion
- modelling the structural behaviour specific to a lexical item:
  - pp-attachment
  - subcategorisation
  - co-ordination

Alternative approach represents features in **DAGs**

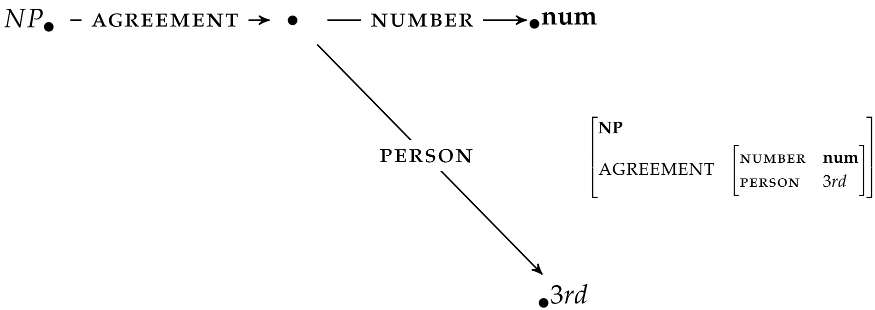
Re-conceptualise words, non-terminal nodes and parses as **Directed Acyclic Graphs** which may be represented as **Attribute Value Matrices**



$$\begin{bmatrix} \text{CAT} & NP \\ \text{AGREEMENT} & \begin{bmatrix} \text{NUMBER} & sing \\ \text{PERSON} & 3rd \end{bmatrix} \end{bmatrix}$$

We have **atomic categories** at each of the terminal nodes and another **AVM**/**DAG** at all other nodes

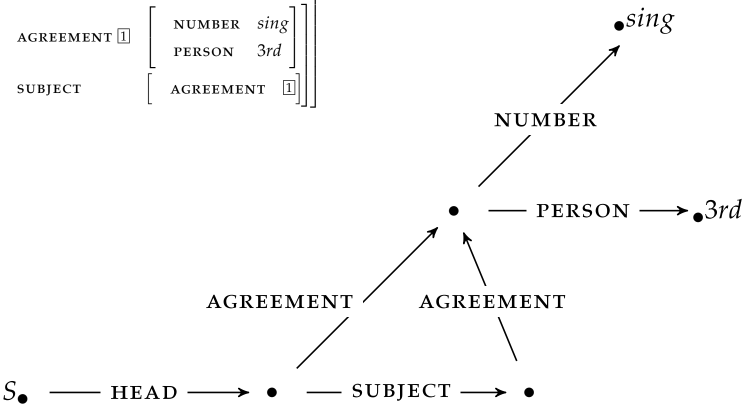
Typing facilitates grammar building. Hierarchies of AVM types can be used to automatically populate features



An shorthand notation uses angle bracket notation to indicate feature paths: e.g. <**NP** AGREEMENT PERSON> would represent the feature path leading to the atomic value *3rd*.

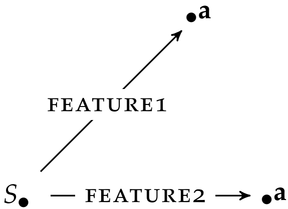
DAGs and AVMs may exhibit **re-entrancy**

$$\begin{bmatrix} \textbf{S} & \\ \textsc{head} & \begin{bmatrix} \textsc{agreement}\ \boxed{1} & \begin{bmatrix} \textsc{number} & \textit{sing} \\ \textsc{person} & \textit{3rd} \end{bmatrix} \\ \textsc{subject} & \begin{bmatrix} \textsc{agreement} & \boxed{1} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$



DAGs and AVMs may exhibit **re-entrancy**

1. Non re-entrant: $\begin{bmatrix} \text{FEATURE1} & \mathbf{a} \\ \text{FEATURE2} & \mathbf{a} \end{bmatrix}$



2. Re-entrant: $\begin{bmatrix} \text{FEATURE1} & \boxed{1}\ \mathbf{a} \\ \text{FEATURE2} & \boxed{1} \end{bmatrix}$

Parsing with DAGs involves **Unification**

- The unification of two DAGs is the most specific DAG which contains all the information in both of the original feature structures.
- Unification fails if the two DAGs contain conflicting information.

Parsing with DAGs involves **Unification**

- The unification of two DAGs is the most specific DAG which contains all the information in both of the original feature structures.
- Unification fails if the two DAGs contain conflicting information.

$$\begin{bmatrix}\textsc{person} & \mathit{3rd}\end{bmatrix} \sqcup \begin{bmatrix}\textsc{number} & \mathit{plural}\end{bmatrix} = \begin{bmatrix}\textsc{person} & \mathit{3rd} \\ \textsc{number} & \mathit{plural}\end{bmatrix}$$

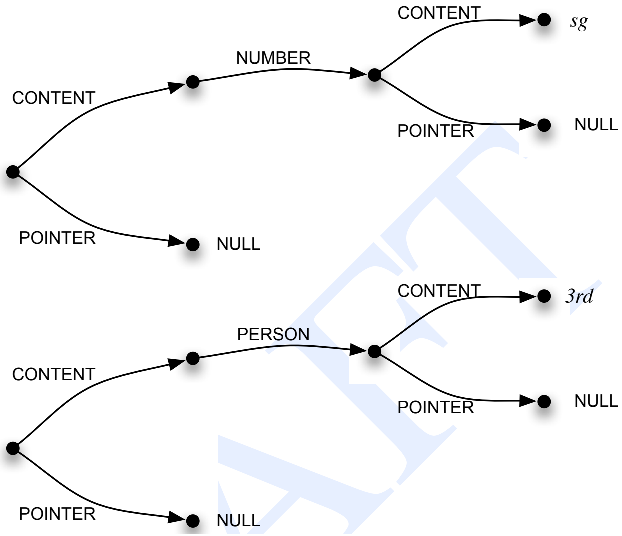
$$\begin{bmatrix}\textsc{person} & \mathit{1st} \\ \textsc{number} & \mathit{plural}\end{bmatrix} \sqcup \begin{bmatrix}\textsc{number} & \begin{bmatrix}\,\end{bmatrix}\end{bmatrix} = \begin{bmatrix}\textsc{person} & \mathit{1st} \\ \textsc{number} & \mathit{plural}\end{bmatrix}$$

$$\begin{bmatrix}\textsc{person} & \mathit{1st} \\ \textsc{number} & \mathit{sing}\end{bmatrix} \sqcup \begin{bmatrix}\textsc{number} & \mathit{plural}\end{bmatrix} = \text{unification fails}$$

$$\begin{bmatrix}\textsc{feature1} & \begin{bmatrix}\textsc{feature2} & \boxed{1}\end{bmatrix} \\ \textsc{feature3}\ \boxed{1} & \end{bmatrix} \sqcup \begin{bmatrix}\textsc{feature3} & \boxed{1}\ a\end{bmatrix} = \begin{bmatrix}\textsc{feature1} & \begin{bmatrix}\textsc{feature2} & a\end{bmatrix} \\ \textsc{feature3} & a\end{bmatrix}$$

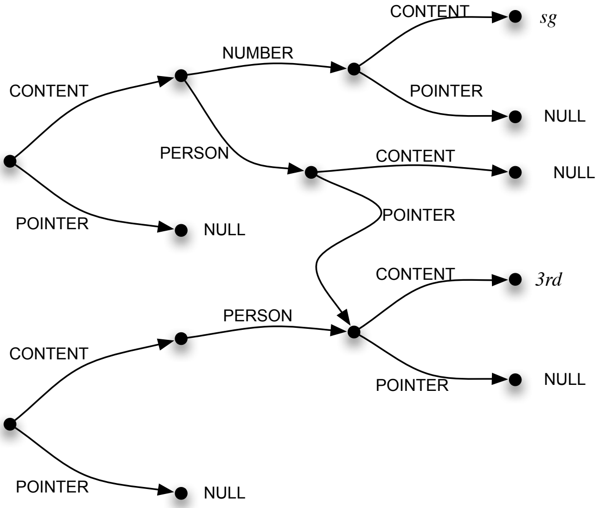
Unification examples in class

Unification **algorithm** requires **extra** graph structure



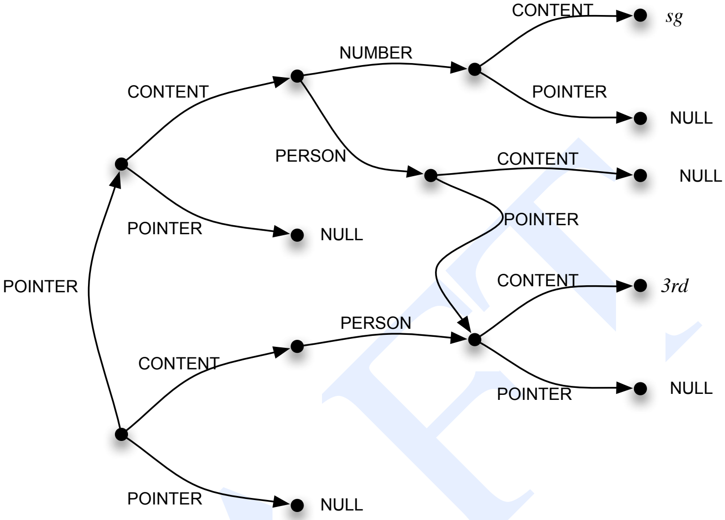
From Jurafsky and Martin version 2

Unification **algorithm** requires **extra** graph structure



From Jurafsky and Martin version 2

Unification **algorithm** requires **extra** graph structure



From Jurafsky and Martin version 2

DAGs can be straighforwardly associated with the lexicon

$$\begin{bmatrix} \textbf{N} & \\ \textsc{agreement} & \begin{bmatrix} \textsc{person} & \textit{3rd} \\ \textsc{number} & \textit{plural} \end{bmatrix} \end{bmatrix} \rightarrow \{\text{fish, rivers, pools, they}\}$$

V $\rightarrow$ {cans, fishes}

<V AGREEMENT PERSON> = *3rd*

<V AGREEMENT NUMBER> = *sing*

$$\left\langle \text{they}, \begin{bmatrix} \textbf{N} & \\ \textsc{agreement} & \begin{bmatrix} \textsc{person} & \textit{3rd} \\ \textsc{number} & \textit{sing} \end{bmatrix} \end{bmatrix} \right\rangle$$

DAGs can be straighforwardly associated with the lexicon

$$\begin{bmatrix} \textbf{N} & \\ \textsc{agreement} & \begin{bmatrix} \textsc{person} & \mathit{3rd} \\ \textsc{number} & \mathit{plural} \end{bmatrix} \end{bmatrix} \rightarrow \{\text{fish, rivers, pools, they}\}$$

V $\rightarrow$ {cans, fishes}

<**V** AGREEMENT PERSON> = *3rd*

<**V** AGREEMENT NUMBER> = *sing*

$$\left\langle \text{they}, \begin{bmatrix} \text{N} & \\ \textsc{agreement} & \begin{bmatrix} \textsc{person} & \mathit{3rd} \\ \textsc{number} & \mathit{sing} \end{bmatrix} \end{bmatrix} \right\rangle$$

DAGs can be straighforwardly associated with the lexicon

$$\begin{bmatrix} \textbf{N} & \\ \textsc{agreement} & \begin{bmatrix} \textsc{person} & \textit{3rd} \\ \textsc{number} & \textit{plural} \end{bmatrix} \end{bmatrix} \rightarrow \{\text{fish, rivers, pools, they}\}$$

$$\text{V} \rightarrow \{\text{cans, fishes}\}$$

$$\langle \textbf{V}\ \textsc{agreement person} \rangle = \textit{3rd}$$

$$\langle \textbf{V}\ \textsc{agreement number} \rangle = \textit{sing}$$

$$\left\langle \text{they}, \begin{bmatrix} \textbf{N} & \\ \textsc{agreement} & \begin{bmatrix} \textsc{person} & \textit{3rd} \\ \textsc{number} & \textit{sing} \end{bmatrix} \end{bmatrix} \right\rangle$$

We can **modify** CFG algorithms to **parse** with DAGs

- We can use any CFG parsing algorithm if:
- associate feature constraints with CFG rules
- unify DAGs in the states
- $S \rightarrow NP\ VP$

  $< NP$ HEAD AGREEMENT $>=< VP$ HEAD AGREEMENT $>$

  $< S$ HEAD $>=< VP$ HEAD $>$
- We would have items like $[X, [0, m], DAG]$ on the agenda or at each cell

Subcategorization is captured by the feature constraints

$$\begin{bmatrix}\textbf{S}\\ \textsc{head}\ \boxed{1}\end{bmatrix} \rightarrow \boxed{2}\begin{bmatrix}\textbf{NP}\\ \textsc{head}\ \begin{bmatrix}\textsc{agreement}\ \boxed{3}\end{bmatrix}\end{bmatrix}\begin{bmatrix}\textbf{VP}\\ \textsc{head}\ \boxed{1}\ \begin{bmatrix}\textsc{agreement}\ \boxed{3}\end{bmatrix}\\ \textsc{subj}\ \boxed{2}\end{bmatrix}$$

$$\begin{bmatrix}\textbf{VP}\\ \textsc{head}\ \boxed{1}\\ \textsc{subj}\ \boxed{3}\end{bmatrix} \rightarrow \begin{bmatrix}\textbf{V}\\ \textsc{head}\ \boxed{1}\\ \textsc{obj}\ \boxed{2}\\ \textsc{subj}\ \boxed{3}\end{bmatrix}\ \boxed{2}\begin{bmatrix}\ \end{bmatrix}$$

Subcategorization is captured by the feature constraints

$$\begin{bmatrix} \textbf{S} \\ \textsc{head}\ \boxed{1} \end{bmatrix} \rightarrow \boxed{2}\begin{bmatrix} \textbf{NP} \\ \textsc{head}\ \begin{bmatrix} \textsc{agreement}\ \boxed{3} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \textbf{VP} \\ \textsc{head}\ \boxed{1}\ \begin{bmatrix} \textsc{agreement}\ \boxed{3} \end{bmatrix} \\ \textsc{subj}\ \boxed{2} \end{bmatrix}$$

$$\begin{bmatrix} \textbf{VP} \\ \textsc{head}\ \boxed{1} \\ \textsc{subj}\ \boxed{3} \end{bmatrix} \rightarrow \begin{bmatrix} \textbf{V} \\ \textsc{head}\ \boxed{1} \\ \textsc{obj}\ \boxed{2} \\ \textsc{subj}\ \boxed{3} \end{bmatrix} \boxed{2}\begin{bmatrix} \ \end{bmatrix}$$

$$\left\langle \text{can}, \begin{bmatrix} \textbf{V} \\ \textsc{head}\ \begin{bmatrix} \textsc{agreement}\ \begin{bmatrix} \ \end{bmatrix} \end{bmatrix} \\ \textsc{obj}\ \begin{bmatrix} \textbf{VP} \\ \textsc{head}\ \begin{bmatrix} \ \end{bmatrix} \end{bmatrix} \\ \textsc{subj}\ \begin{bmatrix} \textbf{NP} \\ \textsc{head}\ \begin{bmatrix} \ \end{bmatrix} \end{bmatrix} \end{bmatrix} \right\rangle$$

Subcategorization is captured by the feature constraints

$$\begin{bmatrix}\textbf{S} & \\ \textsc{head} & \boxed{1}\end{bmatrix} \rightarrow \boxed{2}\begin{bmatrix}\textbf{NP} & \\ \textsc{head} & \begin{bmatrix}\textsc{agreement} & \boxed{3}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\textbf{VP} & \\ \textsc{head} & \boxed{1}\begin{bmatrix}\textsc{agreement} & \boxed{3}\end{bmatrix} \\ \textsc{subj} & \boxed{2}\end{bmatrix}$$

$$\begin{bmatrix}\textbf{VP} & \\ \textsc{head} & \boxed{1} \\ \textsc{subj} & \boxed{3}\end{bmatrix} \rightarrow \begin{bmatrix}\textbf{V} & \\ \textsc{head} & \boxed{1} \\ \textsc{obj} & \boxed{2} \\ \textsc{subj} & \boxed{3}\end{bmatrix} \boxed{2}\begin{bmatrix}\ \end{bmatrix}$$

$$\left\langle \text{can}, \begin{bmatrix}\textbf{V} & \\ \textsc{head} & \begin{bmatrix}\textsc{agreement} & \begin{bmatrix}\ \end{bmatrix}\end{bmatrix} \\ \textsc{obj} & \begin{bmatrix}\textbf{VP} & \\ \textsc{head} & \begin{bmatrix}\ \end{bmatrix}\end{bmatrix} \\ \textsc{subj} & \begin{bmatrix}\textbf{NP} & \\ \textsc{head} & \begin{bmatrix}\ \end{bmatrix}\end{bmatrix}\end{bmatrix} \right\rangle \left\langle \text{cans}, \begin{bmatrix}\textbf{V} & \\ \textsc{head} & \begin{bmatrix}\textsc{agreement} & \begin{bmatrix}\textsc{person} & \textit{3rd} \\ \textsc{number} & \textit{sing}\end{bmatrix}\end{bmatrix} \\ \textsc{obj} & \begin{bmatrix}\textbf{NP} & \\ \textsc{head} & \begin{bmatrix}\ \end{bmatrix}\end{bmatrix} \\ \textsc{subj} & \begin{bmatrix}\textbf{NP} & \\ \textsc{head} & \begin{bmatrix}\ \end{bmatrix}\end{bmatrix}\end{bmatrix} \right\rangle$$

Alternatively use **unification as the parsing operation**

Alternatively use **unification as the parsing operation** instead of just for search-space reduction through feature constraining:

- $X_0 \rightarrow X_1 X_2$

  $< X_1$ HEAD AGREEMENT $>=< X_2$ HEAD AGREEMENT $>$

  $< X_0$ HEAD $>=< X_1$ HEAD $>$

- $X_0 \rightarrow X_1 X_2$

  $< X_0$ HEAD $>< X_1$ HEAD $>$

  $< X_2$ CAT $>= PP$

- $X_0 \rightarrow X_1$ *and* X_2

  $< X_0$ CAT $>< X_1$ CAT $>$

  $< X_1$ CAT $>< X_2$ CAT $>$

Alternatively use **unification as the parsing operation** instead of just for search-space reduction through feature constraining:

- $X_0 \rightarrow X_1 X_2$

  $< X_1 \ \textsc{head agreement} >=< X_2 \ \textsc{head agreement} >$

  $< X_0 \ \textsc{head} >=< X_1 \ \textsc{head} >$

- $X_0 \rightarrow X_1 X_2$

  $< X_0 \ \textsc{head} >< X_1 \ \textsc{head} >$

  $< X_2 \ \textsc{cat} >= PP$

- $X_0 \rightarrow X_1 \ and \ X_2$

  $< X_0 \ \textsc{cat} >< X_1 \ \textsc{cat} >$

  $< X_1 \ \textsc{cat} >< X_2 \ \textsc{cat} >$

Alternatively use **unification as the parsing operation**

Alternatively use **unification as the parsing operation** instead of just for search-space reduction through feature constraining:

- $X_0 \rightarrow X_1 X_2$

  $< X_1 \text{ HEAD AGREEMENT } >=< X_2 \text{ HEAD AGREEMENT } >$

  $< X_0 \text{ HEAD } >=< X_1 \text{ HEAD } >$

- $X_0 \rightarrow X_1 X_2$

  $< X_0 \text{ HEAD } >< X_1 \text{ HEAD } >$

  $< X_2 \text{ CAT } >= PP$

- $X_0 \rightarrow X_1 \textit{ and } X_2$

  $< X_0 \text{ CAT } >< X_1 \text{ CAT } >$

  $< X_1 \text{ CAT } >< X_2 \text{ CAT } >$

Unification based parsing in the wild...

- Focus on adequacy for a wide range of languages as well as tractable for parsing
- Examples include **Lexical Functional Grammar, LFG** (Bresnan and Kaplan) and **Head-driven Phrase Structure Grammar, HPSG** (Pollard and Sag)
- Grammars tend to incorporate aspects of morphology, syntax and compositional semantics:

$$\left\langle \hat{\ } s, \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \mathbf{N} & \\ \text{AGREEMENT} & pl \end{bmatrix} \end{bmatrix} \right\rangle$$

$$\left\langle \text{fox}, \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \mathbf{N} & \\ \text{AGREEMENT} & [\,] \end{bmatrix} \end{bmatrix} \right\rangle$$

$$\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \mathbf{N} & \\ \text{AGREEMENT} & pl \end{bmatrix} \end{bmatrix} \sqcup \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \mathbf{N} & \\ \text{AGREEMENT} & [\,] \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \mathbf{N} & \\ \text{AGREEMENT} & pl \end{bmatrix} \end{bmatrix}$$

If you are interested see: `http://www.delph-in.net`