Projecting Propbank Roles onto the CCGbank

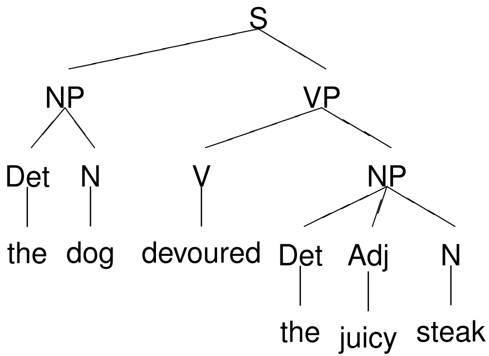
Stephen A. Boxwell
and
Michael White

The Ohio State University

The Penn Treebank WSJ section

- Tens of thousands of sentences from the Wall Street Journal
- Annotated with Part-of-Speech and syntactic structure
- Widely used for a variety of NLP tasks.

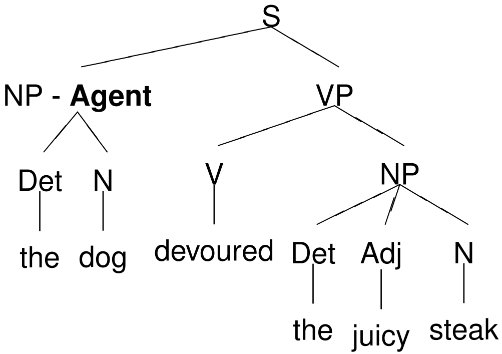
The Penn Treebank WSJ section



The Propbank

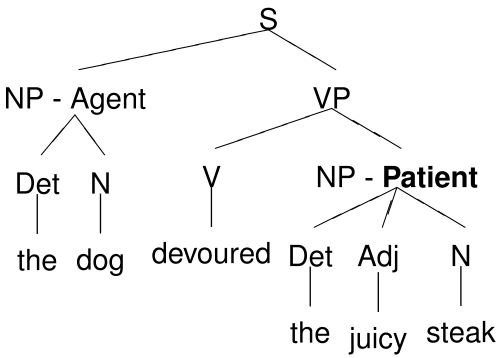
- Annotates semantic roles on Penn Treebank trees
- Distinguishes argument roles from modifier roles (manner of action, duration, etc)
- Identifies role-bearing constituents using terminal index and height
- Example: the "Agent" is at terminal index 2, at height 1

Penn Treebank Tree with Semantic Role annotated



Agent: terminal index 2, height 1

Penn Treebank Tree with Semantic Role annotated



Agent: terminal index 2, height 1
Patient: terminal index 6, height 1

The CCGbank

- Combinatory Categorial Grammar is a grammar formalism that treats words as functions and arguments
- A corpus of CCG derivations derived automatically from the Penn Treebank
- CCGbank removes traces and some punctuation
- CCGbank is binary branching, PTB is not.

The CCG formalism

- CCG uses syntactically informative lexical categories
- Slash direction (/ or \ ) indicates direction of combinatory potential
  - NP/N = determiner (the, a)
  - PP/NP = preposition (to, with)
  - S\NP = intransitive verb (sleep, die)
  - (S\NP)/NP = transitive verb (devour, love)
  - ((S\NP)/NP)/NP = ditransitive verb (give)
  - ((S\NP)/PP)/NP = ditransitive verb (put)

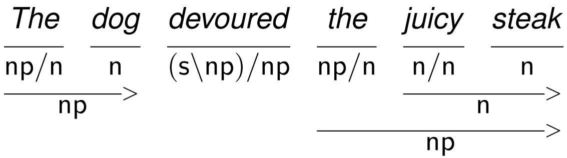
How CCG Categories Make Sentences

The dog devoured the juicy steak

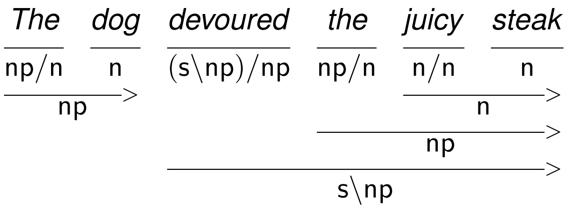
How CCG Categories Make Sentences

$$\begin{array}{cccccc}
\textit{The} & \textit{dog} & \textit{devoured} & \textit{the} & \textit{juicy} & \textit{steak} \\
\hline
\mathrm{np}/\mathrm{n} & \mathrm{n} & (\mathrm{s}\backslash\mathrm{np})/\mathrm{np} & \mathrm{np}/\mathrm{n} & \mathrm{n}/\mathrm{n} & \mathrm{n}
\end{array}$$

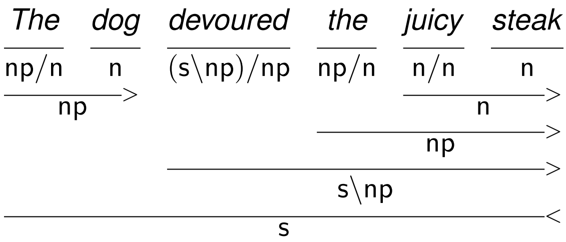
How CCG Categories Make Sentences



How CCG Categories Make Sentences



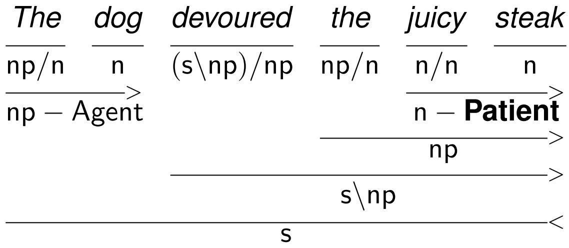
How CCG Categories Make Sentences



The CCGbank and Propbank

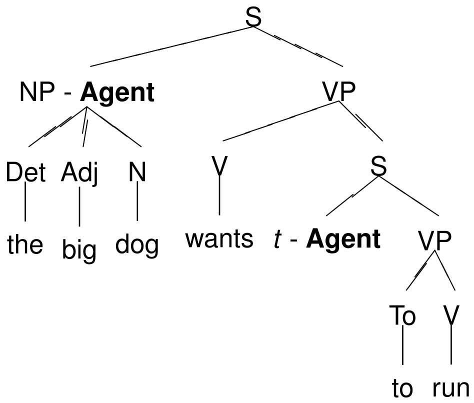
- The CCGbank cannot be used directly with the Propbank
- CCGbank terminals $\neq$ PTB terminals
- Binary branching constraint causes tree height mismatch

Inadvisable Application of Propbank Role to Derivation

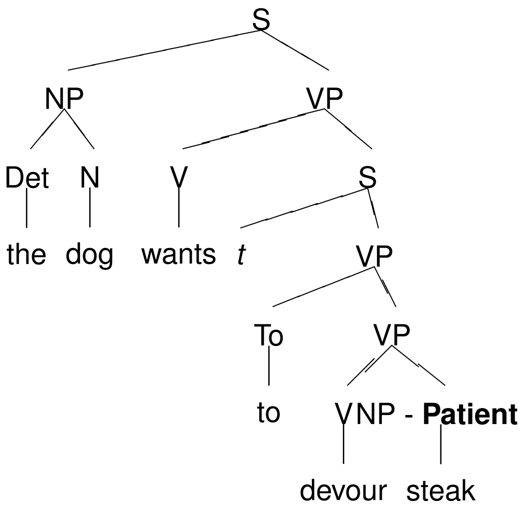


Agent: terminal index 2, height 1
Patient: terminal index 6, height 1

Trace Annotated with Semantic Role



Agent (run): index 3, height 1 AND terminal index 5, height 0



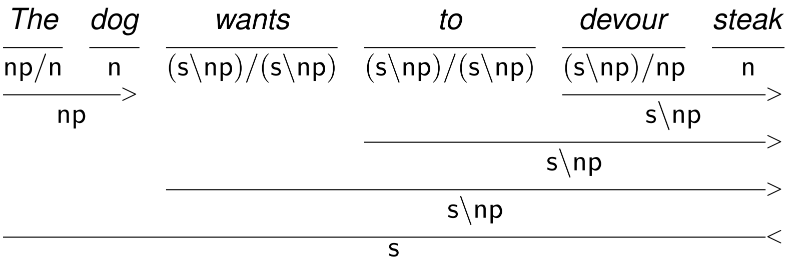
Patient (devour): terminal index 7, height 1

Application of Propbank Role to Derivation Impossible

$$\begin{array}{ccccccc}
\textit{The} & \textit{dog} & \textit{wants} & \textit{to} & \textit{devour} & \textit{steak} & \\
\hline
np/n & n & (s\backslash np)/(s\backslash np) & (s\backslash np)/(s\backslash np) & (s\backslash np)/np & n & \\
\cline{1-2}\cline{5-6}
\multicolumn{2}{c}{np} & & & \multicolumn{2}{c}{s\backslash np} & > \\
\cline{4-6}
 & & & \multicolumn{3}{c}{s\backslash np} & > \\
\cline{3-6}
 & & \multicolumn{4}{c}{s\backslash np} & > \\
\cline{1-6}
\multicolumn{6}{c}{s} & <
\end{array}$$

Patient (devour): terminal index 7, height 1

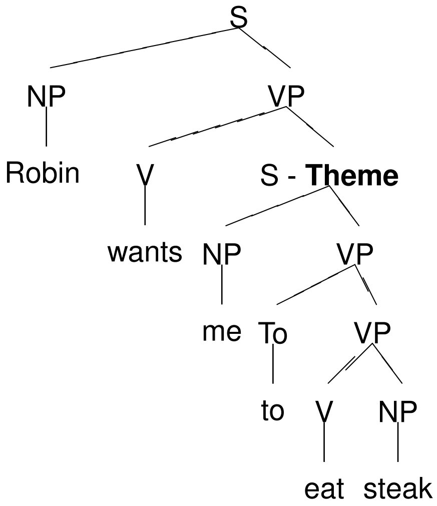
Application of Propbank Role to Derivation Impossible



Patient (devour): terminal index 7, height 1 **FAIL**

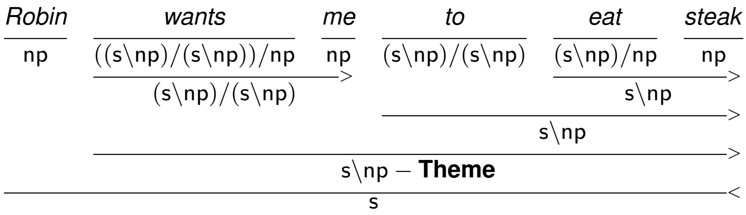
Aligning the CCGbank and Propbank

- Use a minimum edit distance utility to align the terminals of PTB and CCGB
- Create a mapping of PTB terminals to CCGB terminals
- Find a node in the CCG derivation that covers all and only the correct terminals



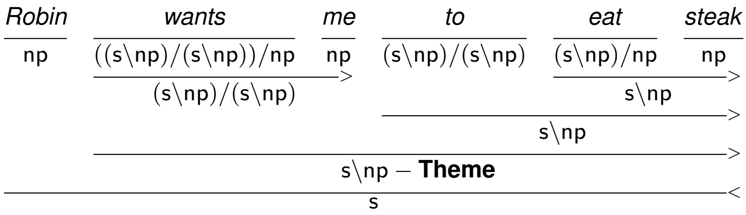
Theme (want): terminal index 3, height 1

Incorrect Application of Semantic Role to Derivation



Theme (want): terminal index 3, height 1

Incorrect Application of Semantic Role to Derivation

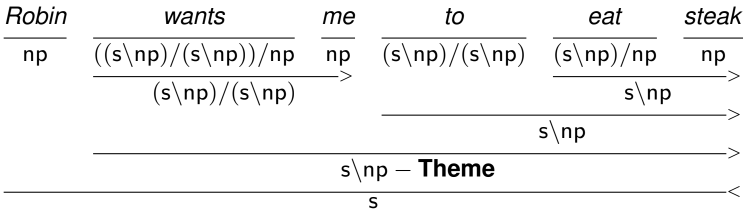


Theme (want): terminal index 3, height 1 **FAIL**

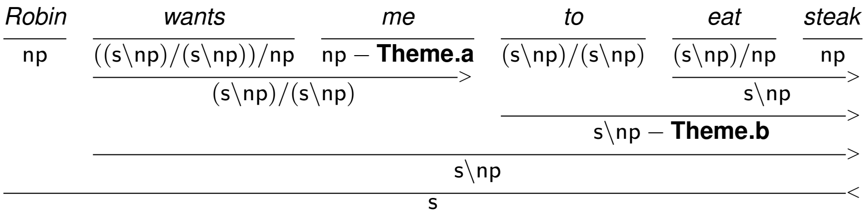
Addressing the Small Clause Mismatch

- Split the role marked on the small clause in two
- Theme → Theme.a, Theme.b
- New notation allows original annotation to be recovered if desired

Incorrect annotation of theme of "wants"



Modified annotation of theme of "wants"



The Argument-Adjunct Distinction

- Penn Treebank does not make a strong distinction between arguments and adjuncts
- Argument - adjunct distinction can make a big difference in word-word dependencies, which has implications for generation and semantic role prediction
- CCG theory requires distinction

A Verb Consuming an Argument

$$
\begin{array}{ccc}
\textit{Shakespeare} & \textit{wrote} & \textit{Macbeth} \\
\hline
\mathsf{np} & (\mathsf{s}\backslash\mathsf{np})/\mathsf{np} & \mathsf{np} \\
 & \multicolumn{2}{c}{\overline{\qquad\qquad\qquad\qquad}>} \\
 & \multicolumn{2}{c}{\mathsf{s}\backslash\mathsf{np}} \\
\multicolumn{3}{c}{\overline{\qquad\qquad\qquad\qquad\qquad\qquad\qquad}<} \\
\multicolumn{3}{c}{\mathsf{s}}
\end{array}
$$

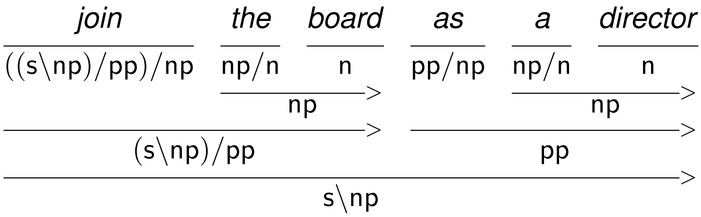
A Verb Modified by an Adjunct

$$\begin{array}{ccccc}
\textit{Shakespeare} & \textit{wrote} & \textit{in} & \textit{1605} & \\
\cline{1-1}\cline{2-2}\cline{3-3}\cline{4-4}
np & s\backslash np & ((s\backslash np)\backslash(s\backslash np))/np & np & \\
\cline{3-4}
 & & \multicolumn{2}{c}{(s\backslash np)\backslash(s\backslash np)} & > \\
\cline{2-4}
 & \multicolumn{3}{c}{s\backslash np} & < \\
\cline{1-4}
\multicolumn{4}{c}{s} & <
\end{array}$$

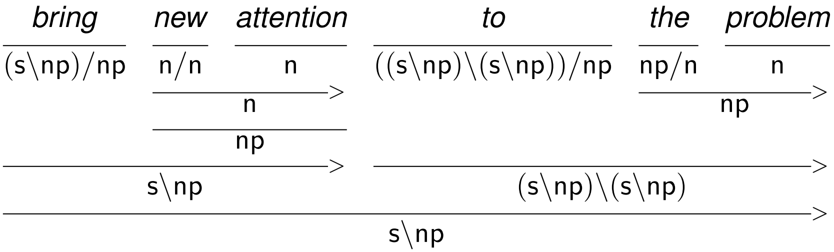
The Argument-Adjunct Distinction

- Because PTB does not make a good distinction between arguments and adjuncts, CCGbank must make its best guess
- Sometimes CCGbank gets it wrong
- These errors can be identified by discrepencies between Propbank roles and CCGbank categories

An Argument that should be an Adjunct



An Adjunct that should be an Argument

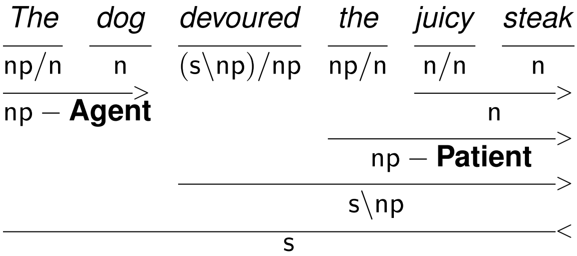


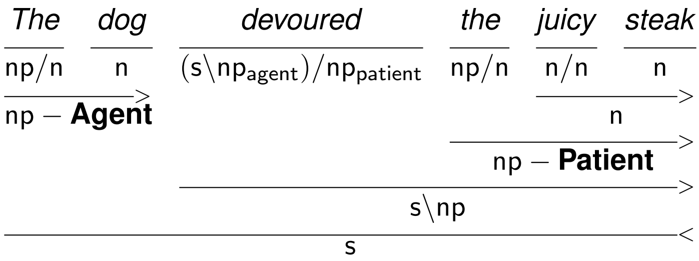
Repairing the CCGbank

- 11569 adjuncts converted to arguments
- 1543 arguments converted to adjuncts
- Modifications reflect the judgement of propbank annotators rather than educated guesses from automatic CCGbank generation algorithm

Why Is This Useful?

- We can use syntactic dependencies to annotate verbal categories with semantic roles
- Creates a mapping from CCG syntactic categories to semantic role frames
- Strong implications for semantic role labeling


The *dog* *devoured* *the* *juicy* *steak*
np/n n (s\np)/np np/n n/n n
>
np – **Agent**
>
n
>
np – **Patient**
>
s\np
<
s


The
dog
devoured
the
juicy
steak
np/n
n
(s\npagent)/nppatient
np/n
n/n
n
>
np – Agent
>
n
>
np – Patient
>
s\np
<
s

How Argument / Adjunct Repair Improves Performance

- 96.85% of syntactic arguments found a numbered role (up from 96.13%)
- 89.24% of semantic roles found a syntactic argument (up from 85.71%)
- differences in improvement reflect the relative number of arguments that are converted to adjuncts, and vice versa.

Current and Future Work

- Current work using the modified CCGbank:
  - Hypertagging - generating surface realizations from a logical form (Espinosa, White, and Mehay, ACL 2008)
  - More precise punctuation analysis for CCG realization (White and Rajkumar)

Acknowledgements

We would like to thank Julia Hockenmaier for the use of her predicate-argument generation tool for CCG derivations. We would also like to thank Chris Brew, Detmar Meurers, Eric Fosler-Lussier, Bob Levine, and Dennis Mehay for their guidance and helpful comments on this work.