

# Projecting Propbank roles onto the CCGbank

Michael White and Chris Brew<sup>1</sup>

Department of Linguistics  
The Ohio State University

OSU Mini-Institute  
Corpus-Based Computational Linguistics  
Day 5

---

<sup>1</sup>Joint work with Stephen Boxwell. (Thanks Steve!)

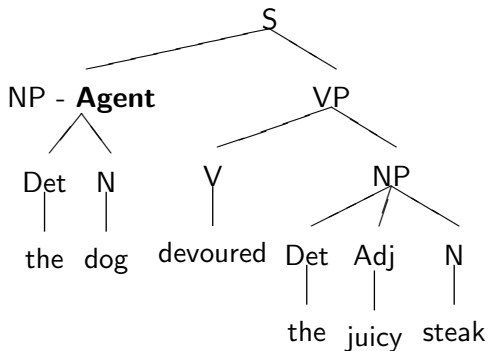
## Onward from the PTB

- Today: Projecting Propbank roles onto the CCGbank [Boxwell and White, 2008]
- Also of note:
  - Adding structure to NPs [Vadas and Curran, 2007]
  - Handling multiword expressions [Hogan et al., 2007]
  - Making punctuation more precise [White and Rajkumar, 2008]
- Not to mention Nombank, Penn Discourse Treebank, Timebank, ...

# Propbank

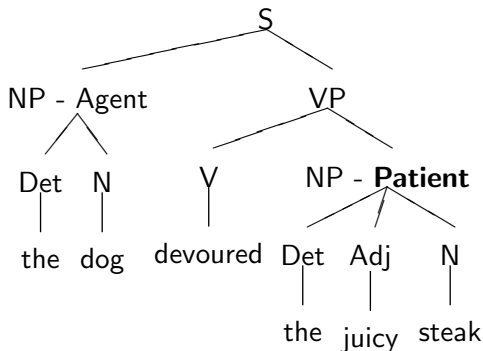
- Annotates semantic roles on Penn Treebank trees
- Distinguishes argument roles from modifier roles (manner of action, duration, etc)
- Identifies role-bearing constituents using terminal index and height
- Example: the “Agent” of *devour* is at terminal index 2, at height 1
  - <http://verbs.colorado.edu/framesets/devour-v.html>

# 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

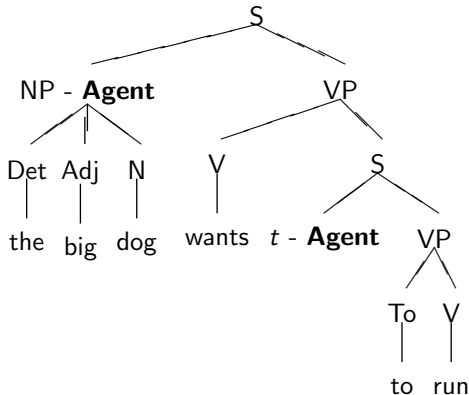
**Patient:** terminal index 6, height 1

# CCGbank and Propbank

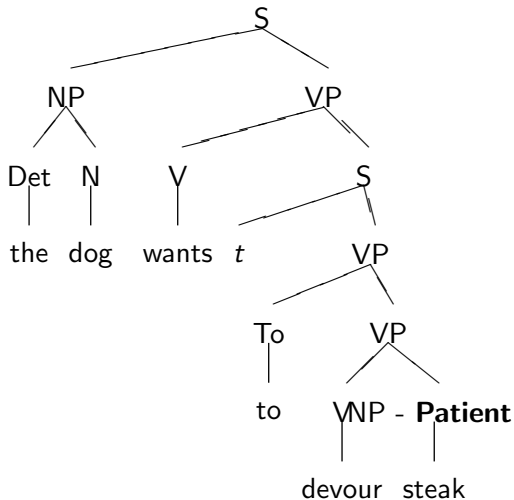
- The CCGbank cannot be used directly with the Propbank
- CCGbank terminals  $\neq$  PTB terminals
- Binary branching constraint causes tree height mismatch



## Trace Annotated with Semantic Role

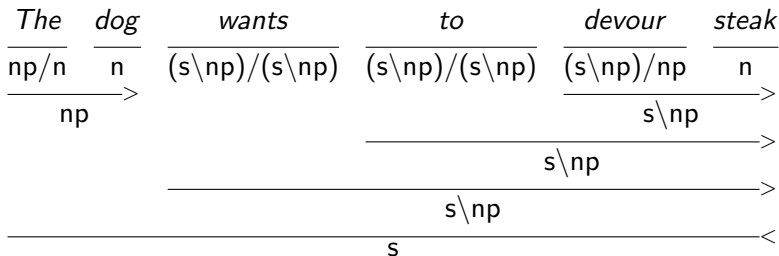


**Agent** (run): index 3, height 1 AND terminal index 5, height 0



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

# Application of Propbank Role to Derivation Impossible

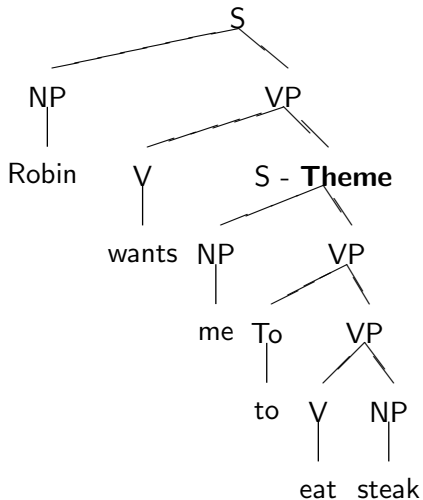


Patient (devour): terminal index 7, height 1



## Aligning the CCGbank and Propbank

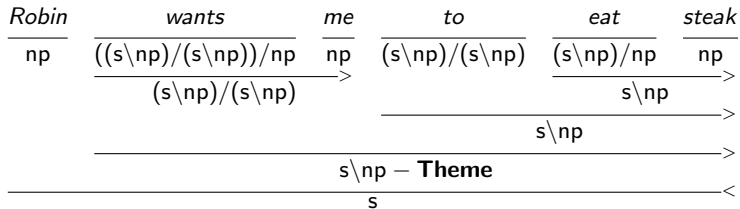
- Use a minimum edit distance utility to align the terminals of PTB and CCGB
- Create a mapping of PTB terminals to CCGB terminals
  - **NB:** This wouldn't be necessary with stand-off annotation!
- Find a node in the CCG derivation that covers all and only the correct terminals



Theme (want): terminal index 3, height 1



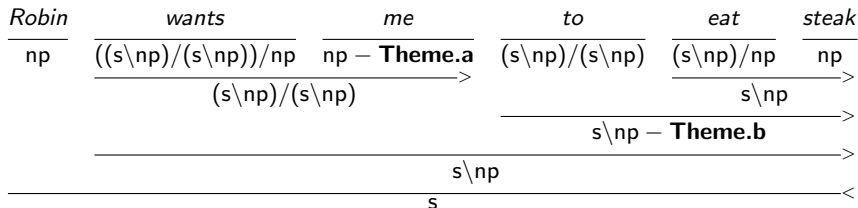
# Incorrect Application of Semantic Role to Derivation



## Addressing the Small Clause Mismatch

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

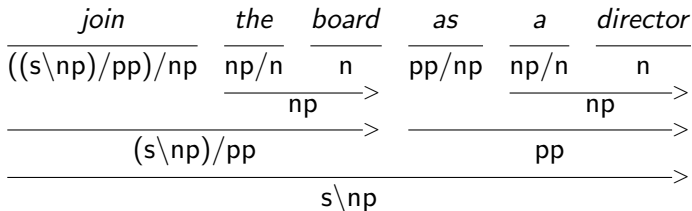
# Modified annotation of theme of “wants”



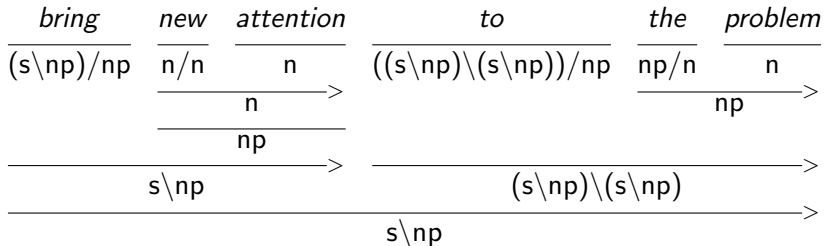
# The Argument-Adjunct Distinction

- 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 discrepancies between Propbank roles and CCGbank categories

## An Argument that should be an Adjunct



# An Adjunct that should be an Argument

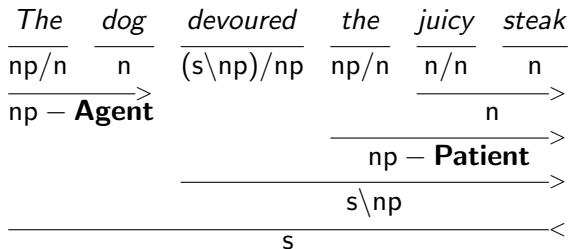


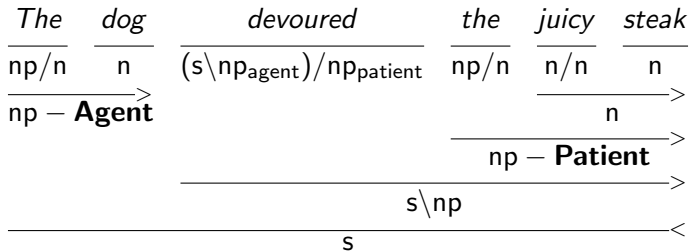
# Repairing the CCGbank

- 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

## Back to the Verb

- We can use syntactic dependencies (even long-distance ones) to annotate verbal categories with semantic roles
- This creates a mapping from CCG lexical categories to semantic role frames









## 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

- Hypertagging, or supertagging for surface realization [Espinosa et al., 2008]
- Supertagging [Bangalore and Joshi, 1999, Curran et al., 2006] and parsing with integrated semantic roles
  - Parsing accuracy crucial to semantic role labeling
  - Joint role prediction important
  - CCGbank/Propbank alignment has been an obstacle

-  Bangalore, S. and Joshi, A. K. (1999).  
Supertagging: An approach to almost parsing.  
*Computational Linguistics*, 25(2):237–265.
-  Boxwell, S. and White, M. (2008).  
Projecting Propbank roles onto the CCGbank.  
In *Proc. LREC-08*.
-  Curran, J., Clark, S., and Vadas, D. (2006).  
Multi-tagging for lexicalized-grammar parsing.  
*Proceedings of the 21st International Conference on  
Computational Linguistics and the 44th annual meeting of the  
ACL*, pages 697–704.
-  Espinosa, D., White, M., and Mehay, D. (2008).  
Hypertagging: Supertagging for surface realization with CCG.  
In *Proc. ACL-08:HLT*.

 Hogan, D., Cafferkey, C., Cahill, A., and van Genabith, J. (2007).

Exploiting multi-word units in history-based probabilistic generation.

In *Proc. EMNLP-CoNLL-07*.

 Vadas, D. and Curran, J. (2007).

Adding noun phrase structure to the penn treebank.

In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 240–247, Prague, Czech Republic. Association for Computational Linguistics.

 White, M. and Rajkumar, R. (2008).

A more precise analysis of punctuation for broad-coverage surface realization with CCG.

In *Proc. of the Workshop on Grammar Engineering Across Frameworks (GEAF08)*.