Learning to Parse Natural Language Commands to a Robot Control System

Cynthia Matuszek,
Evan Herbst,
Luke Zettlemoyer,
Dieter Fox

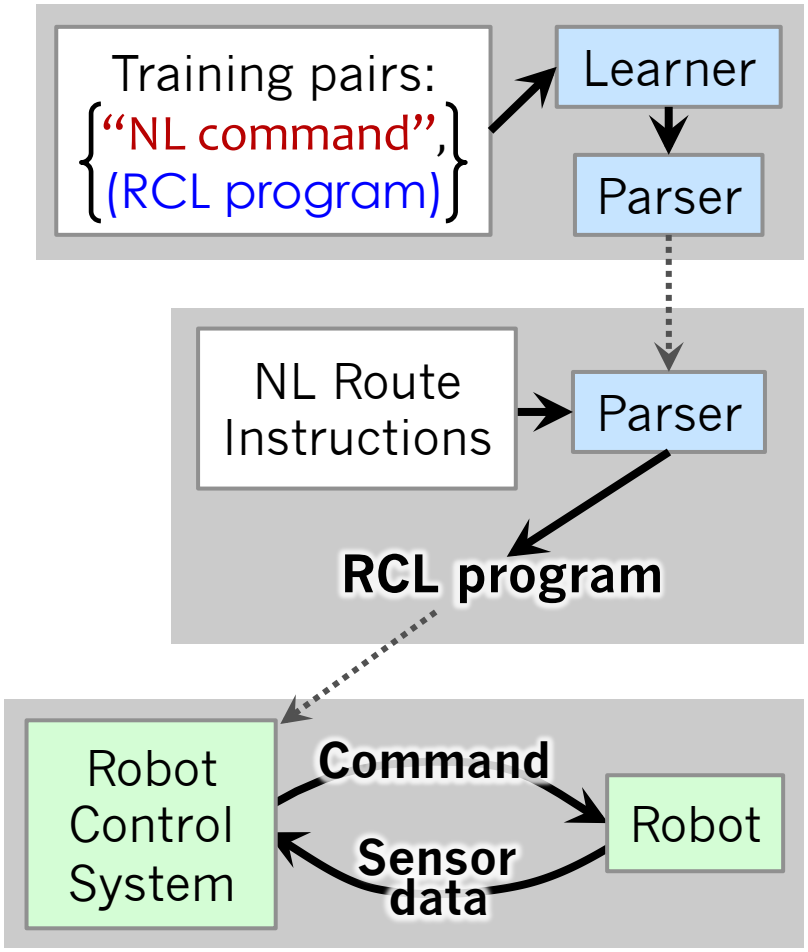
{cynthia|eherbst|lsz|fox}@cs.uw.edu

Motivation

- More natural Human-Robot Interaction (HRI):
  - In the long term, this entails parsing rich human input: speech, gesture, gaze, ...
- Taking instructions from users in **Natural Language**
- Ideally, language understanding **learned** from data
- Key contributions:
  - Follow instructions in a previously unseen world
  - Learn from data to parse natural language
    - Into robot-executable control system

Goal

- "Grounded Language Acquisition"
  - Transform natural language into semantically meaningful representation
  - Map that information to to perceived world
- **Learn a parser**
  - Produces robot-executable commands from NL instructions.



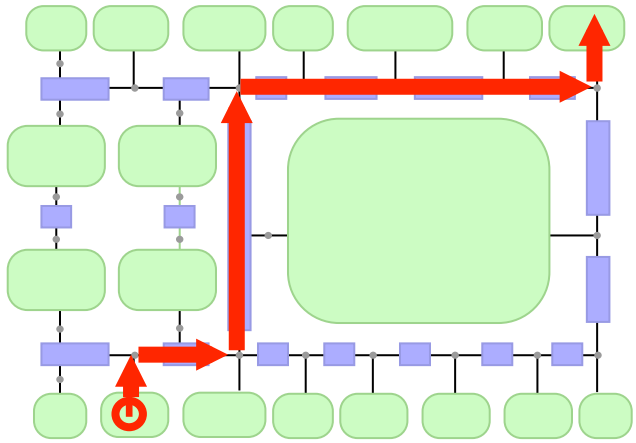
Some Related Work

- Logic-based representations for robot control
  [Beetz *et al*, Konolige *et al*, Kress-Gazit *et al*, Dantam-Stilman *et al*, ...]
- Direction following in rich simulated environments
  [Kuipers-MacMahon-Wong *et al*, Chen-Mooney, ...]
- Learned semantic parsing [Zettlemoyer *et al*, Liang *et al*, ...]
- Learn to parse NLP for RoboCup and direction following (with minimal supervision) [Mooney *et al*]
- Parsing NL in known world and action models: for direction following; for forklift operation
  [Tellex-Kollar-Roy *et al*]

[Matuszek *et al* 2010]

Testbed: Route Instructions

"Leave the room and turn right, take the first left, go past the meeting room and go right, then go to the end of the hall and turn left."



- Previous work grounded instructions directly into the map – no target concepts such as **while**
- Parser must be able to produce many **possible** groundings:
- High-level concepts are worse:
  - "go to the end of the hall,"
  - "keep turning right until you can't any more", ...

"Take the second left."

1: (go (hall) (4junction 1)
(hall) (3junction lt 0) (room))
2: (go (room) (4junction 1)
(room) (3junction lt 0) (room))
3: (go (hall) (4junction 1)
(hall) (3junction rt 1) (room)
(3junction lt 0) (room)) ...

Example Commands

"Go left to the end of the hall."

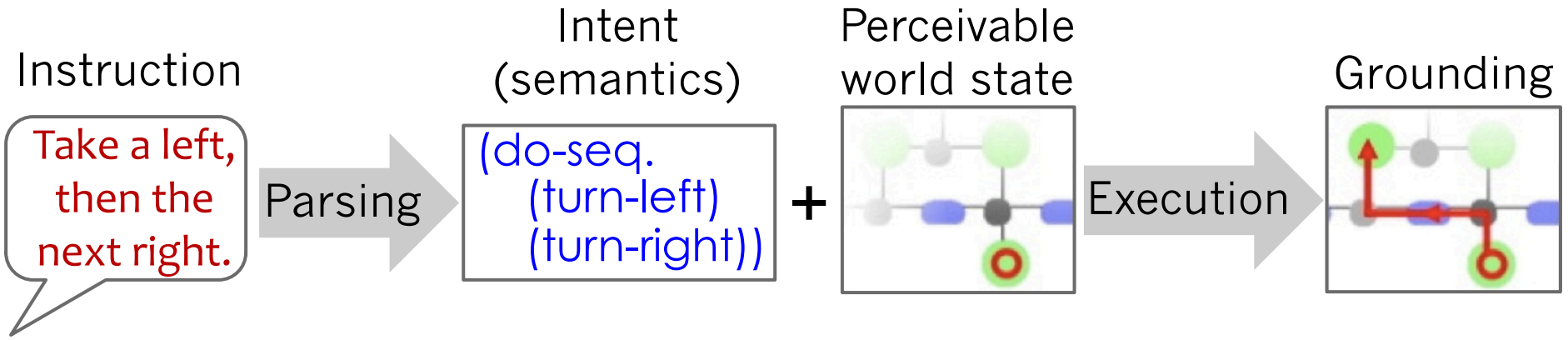
```
(do-sequentially
   (turn-left current-loc)
   (do-until
      (or
         (not (exists forward-loc))
         (room forward-loc))
      (move-to forward-loc)))
```

"Go to the third junction and take a right."

```
(do-sequentially
   (do-n-times 3
      (do-sequentially
         (move-to forward-loc)
         (do-until
            (junction current-loc)
            (move-to forward-loc))))
   (turn-right current-loc))
```

- Humans generate English; parser generates RCL
- Assumptions: robot can execute actions, recognize objects, and determine conditionals
- Primitives can encode complex activities

Approach: Semantic Parsing



- **Parse from NL to a formal control language:** Robot Control Language, or RCL.
- **Train** semantic parsing model
  - → Distribution over RCL sequences for any NL sentence
- Application of learned system: parse new instructions, with simulated agent in unknown map

[Steedman 2000, Clark-Curran 2003]

Categorial Combinatory Grammars

- Capture **syntax** and **semantics** of language
- Parse sentences to expressions in λ-calculus
- Space of possible parses defined by:

lexical entries → go to $\vdash S / NP : \lambda x.move\text{-}to(x)$
lexical entries → junction $\vdash N : \lambda x.junction(x)$

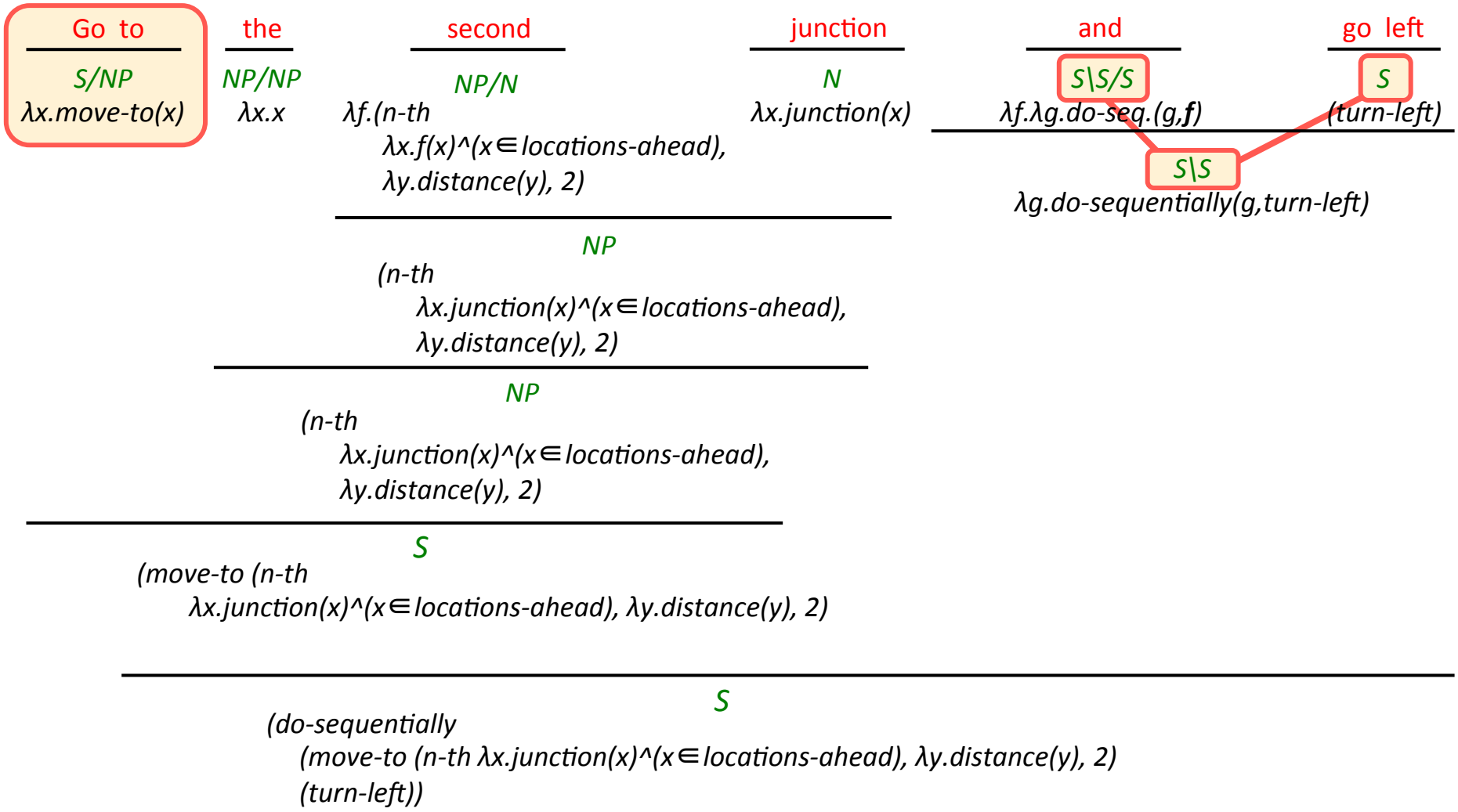
along with combinatory rules.

- Probabilistic CCGs define a log-linear model over:

sentence x
parse y
logical form z

$$p(y,z \mid x;\theta,\Lambda) = \frac{e^{\theta \cdot \phi(x,y,z)}}{\sum_{y',z'} e^{\theta \cdot \phi(x,y',z')}}$$

Example CCG Parse



[Zettlemoyer-Collins 2005, Kwiatkowski *et al* 2010,2011]

Learning Probabilistic CCGs

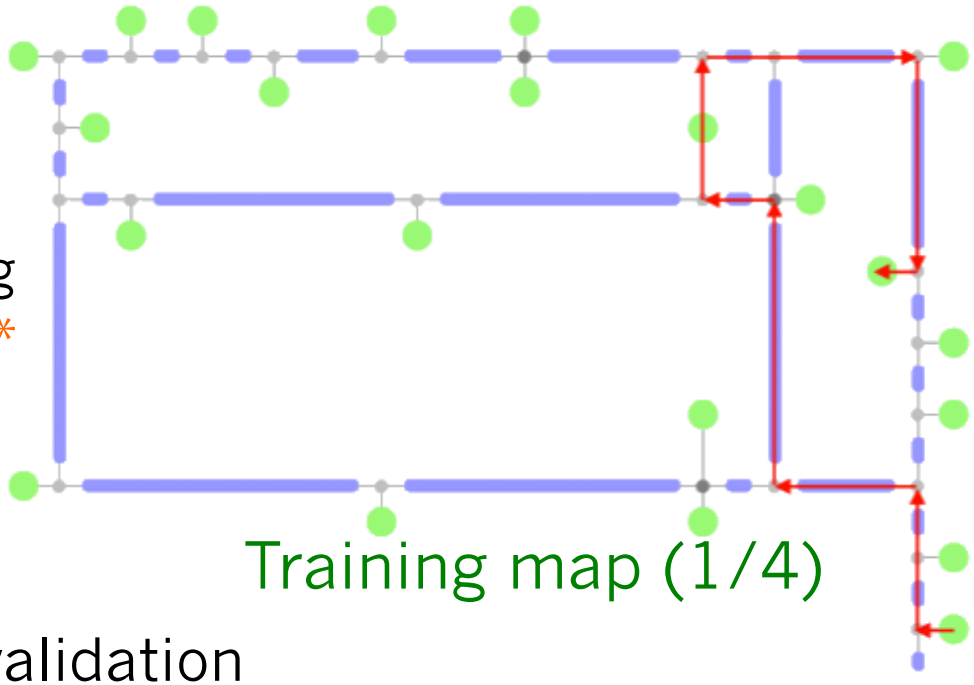
- **Input:** Example pairs of sentences and logical forms
- **Output:** PCCG lexicon and feature weights
- **Structure learning:** Generate lexical items from examples
  - Via combination or splitting rules
- **Data driven updates:** add lexical items only when involved in generating most likely parse of formula
- Parameter estimation via gradient descent

$$\frac{\partial \log(p(z_i \mid x_i;\theta,\Lambda)}{\partial \theta_j} = E_{p(y|x_i,z_i;\Theta,\Lambda)}\left[\phi_j(x_i,y,z_i)\right] - E_{p(y,z|x_i;\Theta,\Lambda)}\left[\phi_j(x_i,y,z)\right]$$

[Friedman *et al*, 2007]

Experimental Setup

- **Training**
  - Route instructions:
    - 9 routes, 2 maps
    - Semantic labeling using Voronoi Random Fields*
  - Annotated in RCL



Training map (1/4)

- **Testing**
  - Parsing: 10-fold cross-validation
  - Navigation
    - 1200 generated routes, 2 novel maps
    - Map discovery simultaneous with following RCL program

Experiment: Parser

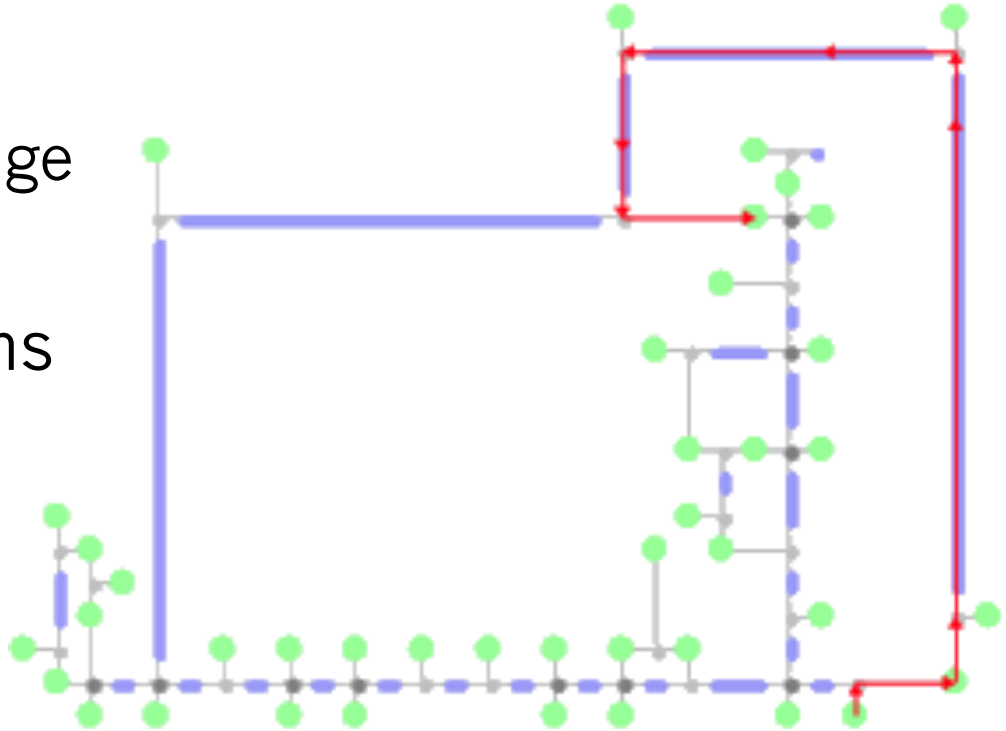
- Route instructions from non-expert users
  - **Segmented** and **annotated** in RCL
- Parser test: 10-fold cross-validation on parsing
- Compare produced parses against gold-standard RCL annotations
- Tests **exact** match only

Precision	Recall	F1-measure
71.0%	72.6%	71.8%

- Evaluates performance on individual sentences, **not** testing full system against a map

Experiment: Navigation

- Route Following with complex language
  - Novel route instructions, novel map
    - 418 sentences total
    - 25 participants
    - Complex language represented
  - Route instructions generated from 2 held-out participants

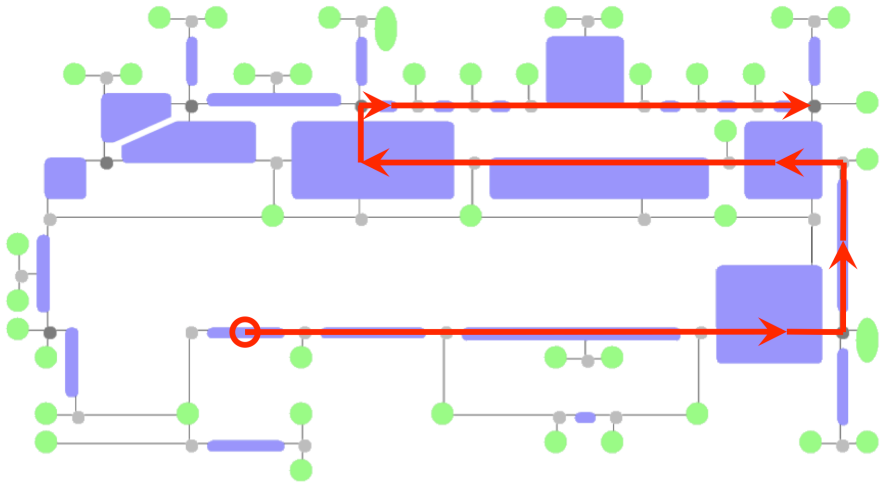


Results

- 10 trials
- Generated:
  - 1,000 short paths
  - 200 long paths
  - Route instructions generated by combining sentences from held-out participants

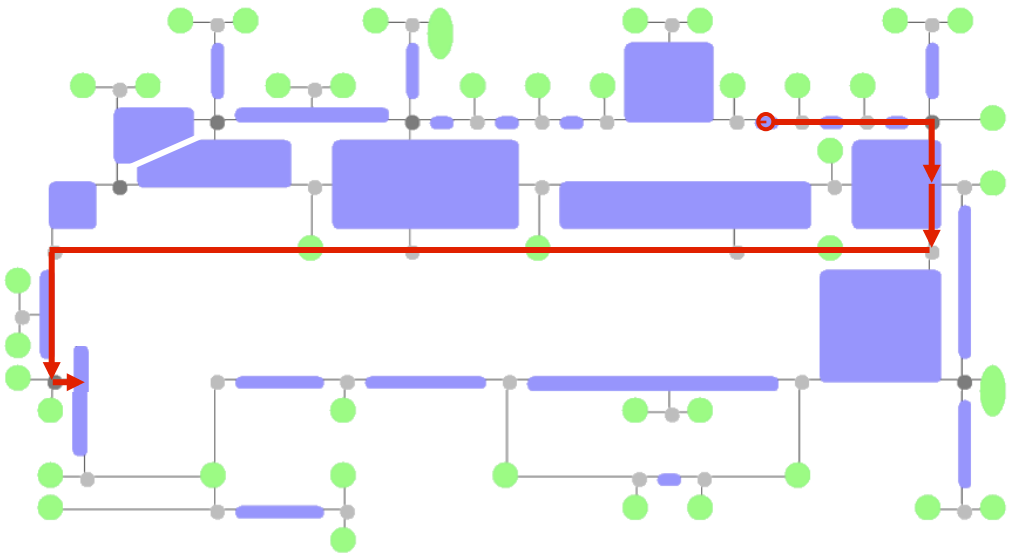
Short	Long
66.3%	48.9%

"Go down the long hallway past three intersections, turn left, take the hallway to your left, go through two intersections, turn right, and go forward to the end of the hall."



Example Parse

- Go past two junctions and turn right, go forward to the 3-way intersection, take the first right, go straight through the second junction then go left, and turn left again.



```
(do-sequentially
  (do-sequentially
    (do-n-times 2
      (do-sequentially
        (do-until
          (junction current-loc)
          (move-to forward-loc))
        (move-to forward-loc)))
    (turn-right current-loc))
  (do-until
    (junction3 current-loc)
    (move-to forward-loc))
  (turn-right current-loc)
  (do-sequentially
    (do-n-times 2
      (do-sequentially
        (do-until
          (junction current-loc)
          (move-to forward-loc))
        (move-to forward-loc)))
    (turn-left current-loc))
  (turn-left current-loc))
```

Conclusions

- **It is possible to combine advanced natural language processing with robotic perception and control.**
  - Parser can be **learned from data** to handle complex, procedural NL for robot instruction
  - Including counting, loops, conditionals, polysemies
- Future Work
  - Local error recovery; more/more varied training data
  - More interesting data
  - Generate ranked list of programs to execute
  - Analyze formal correctness of language; of produced programs
- Other work extends underpinnings of formal language – ICML 2012