Understanding Natural Language Commands for Robotic Navigation and Mobile Manipulation

Stefanie Tellex and Thomas Kollar and Steven Dickerson and Matthew R. Walter and Ashis Gopal Banerjee and Seth Teller and Nicholas Roy

Presented by: Michael Li

*Figures from several papers

Problem Statement

To create a system in which robots can *robustly* follow *spoken instructions* from humans

Commands from the corpus
- Go to the first crate on the left and pick it up.
- Pick up the pallet of boxes in the middle and place them on the trailer to the left.
- Go forward and drop the pallets to the right of the first set of tires.
- Pick up the tire pallet off the truck and set it down

(a) Robotic forklift (b) Sample commands

Figure 1: A target robotic platform for mobile manipulation and navigation (Teller et al. 2010), and sample commands from the domain, created by untrained human annotators. Our system can successfully follow these commands.

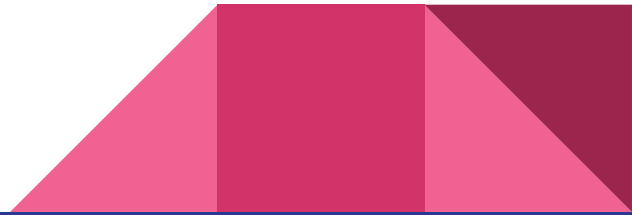
Problem Statement (formally)

- Γ is the set of all groundings
  - Elements denoted as γ_i
  - Real world object connected to a language word
- Φ is the set of all binary correspondence variables
  - Elements denoted as ϕ_i
  - Elements are true if the corresponding γ_i is correct

$$\underset{\Gamma}{\operatorname{argmax}}\, p(\Phi = True | \text{command}, \Gamma)$$

Spatial Description Clauses

- Consists of a figure, relation, and variable number of landmarks
- Matches with part of the input
- Types
  - EVENT: action sequence that takes place (e.g. “Move the tire pallet”)
  - OBJECT: thing in the world (e.g. “the person”, “forklift”, “the truck”)
  - PLACE: place in the world (e.g. “on the truck”)
  - PATH: A path or path fragment through the world (e.g. “past the truck”)
- Automatic SDC extractor - uses Stanford dependencies via Stanford Parser



Spatial Description Clauses Examples

$$EVENT_1(r = \texttt{Put},\ l = OBJ_2(f = \texttt{the pallet}),\ l2 = PLACE_3(r = \texttt{on},\ l = OBJ_4(f = \texttt{the truck})))$$

(a) SDC tree

$$EVENT_1(r = \texttt{Go}\ l = PATH_2(r = \texttt{to},\ l = OBJ_3(f = OBJ_4(f = \texttt{the pallet}),\ r = \texttt{on},\ l = OBJ_5(f = \texttt{the truck}))))$$

(a) SDC tree

Generalized Grounding Graphs

Their contribution: An algorithm to construct bipartite graph is generated based on the linguistic structure defined by a tree of SDCs

$$p(\Phi|\text{commands}, \Gamma) = p(\Phi|\text{SDCs}, \Gamma)$$
$$= \frac{1}{Z} \prod_i \Psi_i(\phi_i, \text{SDC}_i, \Gamma)$$

Generalized Grounding Graphs Terminology

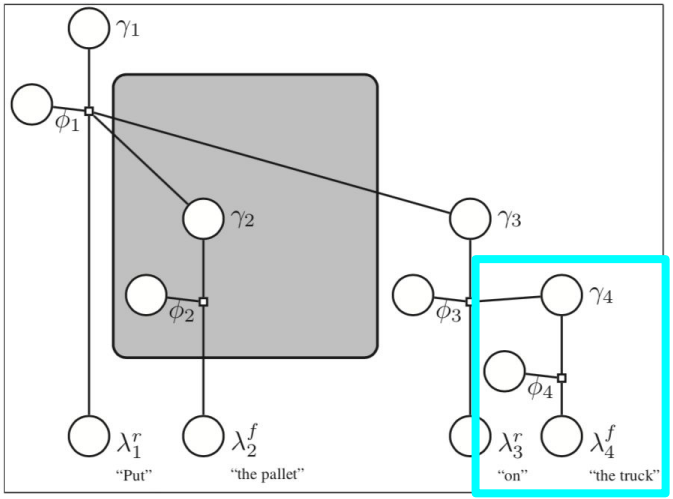
Node types

- Circle (random variables)

ϕ_i True if the grounding γ_i corresponds to i^{th} SDC, and false otherwise.
λ_i^f The words of the figure field of the i^{th} SDC.
λ_i^r The words of the relation field of the i^{th} SDC.
λ_i^{l1}, λ_i^{l2} The words of the first and second landmark fields of the i^{th} SDC; if non-empty, always a child SDC.
$\gamma_i^f, \gamma_i^{l1}, \gamma_i^{l2} \in \Gamma$ The groundings associated with the corresponding field(s) of the i^{th} SDC: the state sequence of the robot (or an object), or a location in the semantic map.

$$EVENT_1(r = \texttt{Put},\ l = OBJ_2(f = \texttt{the pallet}),\ l2 = PLACE_3(r = \texttt{on},\ l = OBJ_4(f = \texttt{the truck})))$$

(a) SDC tree



(b) Induced model

Generalized Grounding Graphs Terminology

Node types

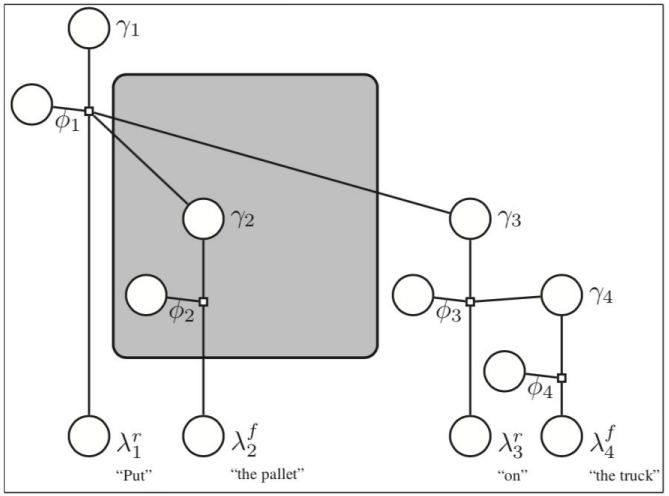
- Square (factor nodes)

$\Psi(\phi_i, \lambda_i^f, \gamma_i)$ for leaf SDCs.

$\Psi(\phi_i, \lambda_i^r, \gamma_i^f, \gamma_i^{l1})$ or $\Psi(\phi_i, \lambda_i^r, \gamma_i^f, \gamma_i^{l1}, \gamma_i^{l2})$ for internal SDCs.

$$EVENT_1(r = \texttt{Put},\ l = OBJ_2(f = \texttt{the pallet}),\ l2 = PLACE_3(r = \texttt{on},\ l = OBJ_4(f = \texttt{the truck})))$$

(a) SDC tree



(b) Induced model

The Model

- Discriminative model, conditional random field (CRF)

$$\Psi_i(\phi_i, \mathrm{SDC}_i, \Gamma) = \exp\left(\sum_k \mu_k s_k(\phi_i, \mathrm{SDC}_i, \Gamma)\right)$$

  - where s_k outputs a binary decision
  - where μ_k is the weight of this binary feature
- Training: Maximum Likelihood, gradient descent
- Inference: Optimize over Γ by fixing $\mathbf{\Phi}$ and SDCs

Features

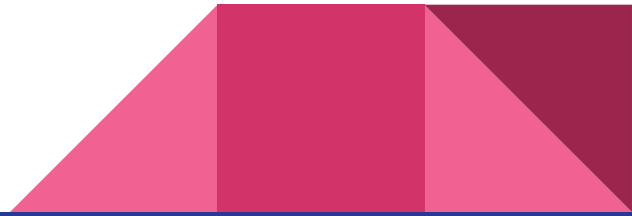
- This CRF requires binary features s_k and weights μ_k
- Base features
  - OBJECT and PLACE SDCs: *supports()*, *distance(), avs()*
  - PATH and EVENT SDCs: distance of path from object, average distance of a path from an object
- Cartesian product of base features with corresponding words in SDC
- Continuous features (e.g. distances) are split into uniform buckets

$$s(\gamma_i^l, \gamma_i^f, \lambda_i^r) \triangleq |x_{\gamma_i^l} - x_{\gamma_i^f}| \wedge (\text{“down from”} \in \lambda_i^r).$$

* figure from Sachithra Hemachandra

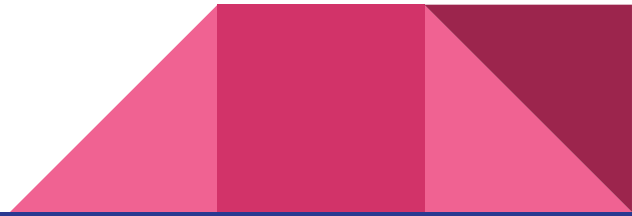
Feature Numbers

- 49 base features for leaf OBJECT and PATH SDCs
- 56 base features for internal OBJECT and PATH SDCs
- 112 base features for EVENT SDCs
- 47 base features for PATH SDCs
- 147274 total binary features after Cartesian product



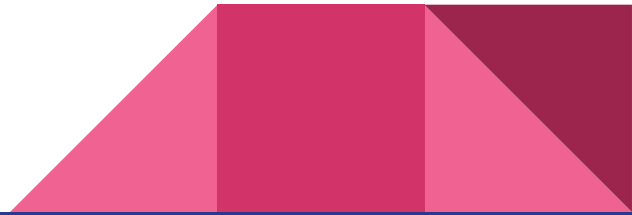
Features

- What about unknown words?
- Calculates word-label similarity
  - WordNet
  - Co-occurrence statistics via Flickr

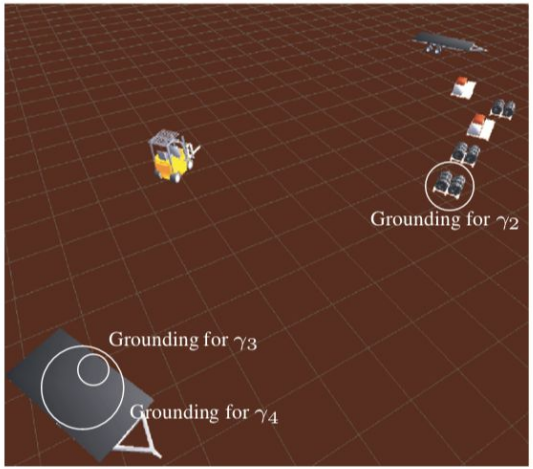


Inference

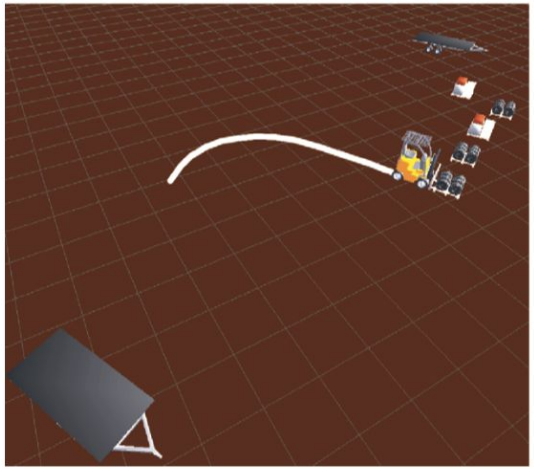
- Given a command, we want the most probable grounding
- Fix Φ and SDCs - search for groundings Γ to maximize the probability of a match
- Two passes
  - First, find and score candidate groundings for OBJECT and PLACE SDCs
  - Beam Search
  - Second, use these candidates to find EVENT and PATH SDCs
- Returns: Object groundings + sequence of actions



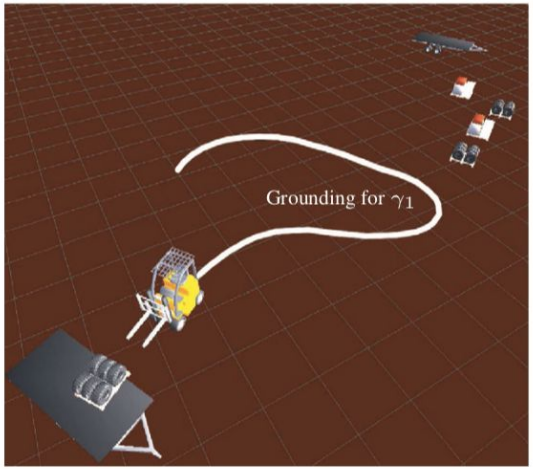
Example Inference



(a) Object groundings



(b) Pick up the pallet

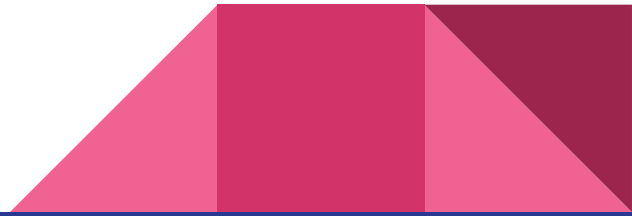


(c) Put it on the truck

Figure 4: A sequence of the actions that the forklift takes in response to the command, "Put the tire pallet on the truck." (a) The search grounds objects and places in the world based on their initial positions. (b) The forklift executes the first action, picking up the pallet. (c) The forklift puts the pallet on the trailer.

Evaluation

- Amazon Mechanical Turk
  - 45 subjects
  - 22 videos
  - Video of a forklift executing an action in a simulated warehouse
  - Gathered 13 commands per video
- Manually annotated associations between SDCs and grounded objects to get initial ground truth data
- Negative examples
  - Random grounding with each SDC



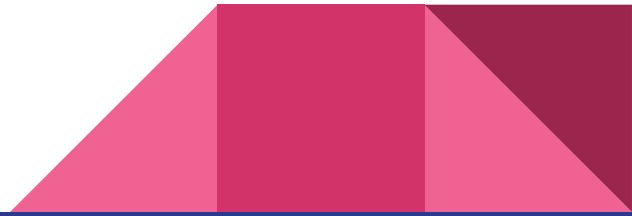
Cost Function Evaluation

SDC type	Precision	Recall	F-score	Accuracy
OBJECT	0.93	0.94	0.94	0.91
PLACE	0.70	0.70	0.70	0.70
PATH	0.86	0.75	0.80	0.81
EVENT	0.84	0.73	0.78	0.80
Overall	0.90	0.88	0.89	0.86

Table 1: Performance of the learned model at predicting the correspondence variable ϕ.

End-to-end Evaluation

- Infer plans given only commands from the test set and a starting location
- Created simulations based on the plans and uploaded to AMT
- Five-point Likert scale with how well the forklift is following the command



End-to-end Evaluation

	Precision
Command with original video	0.91 ($\pm$0.01)
Command with random video	0.11 ($\pm$0.02)

Table 2: The fraction of end-to-end commands considered correct by our annotators for known correct and incorrect videos. We show the 95% confidence intervals in parentheses.

SDC, Probability Distribution Pairing Results

	Precision
Constrained search, random cost	0.28 (±0.05)
Ground truth SDCs (top 30), learned cost	0.63 (±0.08)
Automatic SDCs (top 30), learned cost	0.54 (±0.08)
Ground truth SDCs (all), learned cost	0.47 (±0.04)

Table 3: The fraction of commands considered correct by our annotators for different configurations of our system. We show the 95% confidence intervals in parentheses.

Overview

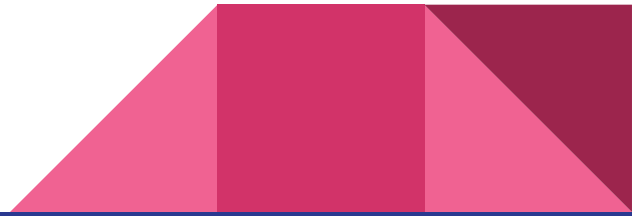
1. $$\underset{\Gamma}{\operatorname{argmax}}\, p(\Phi = True|\text{command}, \Gamma)$$

2. $$p(\Phi|\text{commands}, \Gamma) = p(\Phi|\text{SDCs}, \Gamma)$$
$$= \frac{1}{Z} \prod_i \Psi_i(\phi_i, \text{SDC}_i, \Gamma)$$

3. $$\Psi_i(\phi_i, \text{SDC}_i, \Gamma) = \exp\left(\sum_k \mu_k s_k(\phi_i, \text{SDC}_i, \Gamma)\right)$$

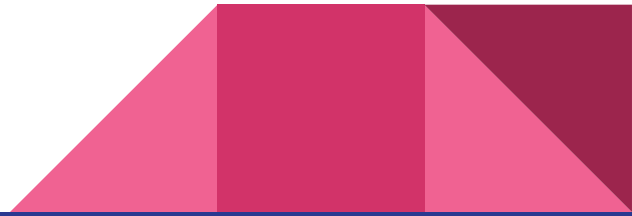
Conclusion

- The authors have presented a system to generate actions via a probabilistic graphical model structured by natural language
- Optimized a CRF for the best groundings



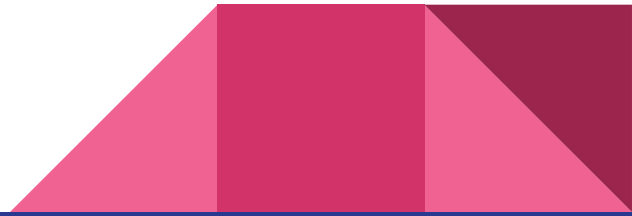
Critique

- Very small environment/world
- Relatively small data set
- Hand labeled features
- CRF seems to have a lot of unused/useless features
  - By product of the Cartesian Product, but maybe can prune even more
- No mention of how long it takes for the system to calculate the set of actions



Future Work

- Evaluate and test on longer commands
  - Disambiguate ambiguous pronouns
- Extend the system to clarify ambiguous portions of the command
  - Model it
  - Ask clarifying questions



Questions?

Citations

- Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew R. Walter, Ashis Gopal Banerjee, Seth Teller, Nicholas Roy, Understanding Natural Language Commands for Robotic Navigation and Mobile Manipulation, In Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI), August 2011.
- Some slides adapted from: https://bcourses.berkeley.edu/courses/1464824/files/72005575/download?verifier=knnqpxI3WOEAWIGncpsfIIggFwsPGFS96FNMTdaf&wrap=1
- Sachithra Hemachandra, Learning semantic maps from natural language, PhD Thesis, https://dspace.mit.edu/handle/1721.1/97757
-