Probabilistic Models for Real-Time Natural Language Corrections to Assistive Robotic Manipulators

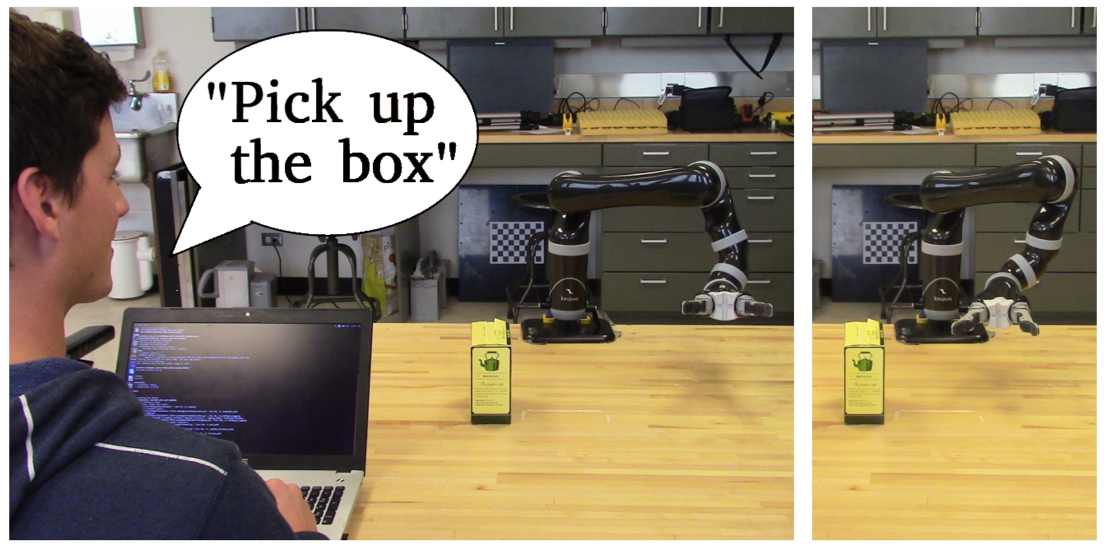
Jacob Arkin[1], Alex Broad[2], Thomas Howard[1], Brenna Argall[2]

[1]Hajim School of Engineering and Applied Sciences, University of Rochester, Rochester, NY 14627

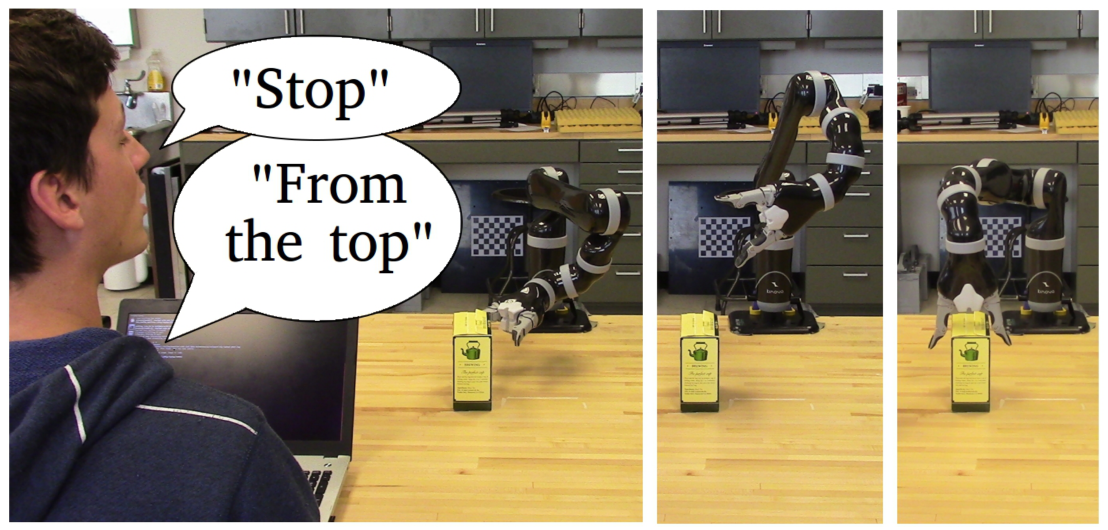
[2]McCormick School of Engineering, Northwestern University, Evanston, IL 60208

Abstract

Natural language interfaces offer an intuitive and efficient means of communication for cooperative human-robot teams in a variety of collaborative domains, such as rehabilitative and assistive robotics. In particular, this unique modality can allow users to customize the execution of certain tasks (Fig. 1). We explore a variation of Distributed Correspondence Graphs (DCG) [2] that provides the capacity to interject corrections to the trajectory of a robotic manipulator in real-time.



(a) Initial command.



(b) Corrective instruction.

Figure 1. A robotic manipulator executing a command given by a person (a), followed by a natural language correction (b).

Contributions

Historically, these models have treated instructions as independent utterances; we introduce the notion of prior context, consisting of the initially expressed instruction and its grounded root phrase. We have implemented and demonstrated a probabilisitic model for real-time natural language corrections on assistive robotic manipulators.

Technical Approach

We seek to find the most likely robot trajectory $(\mathbf{x_i}^*(t))$ described by a natural language correction (Λ) in the context of the perceived environment (Υ) and the prior robotic behavior $(\mathbf{x_{i-1}}(t))$:

$$\mathbf{x_i}^*(t) = \arg\max_{\mathbf{x_i}(t)} p\left(\mathbf{x_i}(t) \,|\, \Lambda, \Upsilon, \mathbf{x_{i-1}}(t)\right) \qquad (1)$$

We first generate a parse tree of the correction (Fig. 2), which is then used to provide the structure of the DCG (Fig. 3).

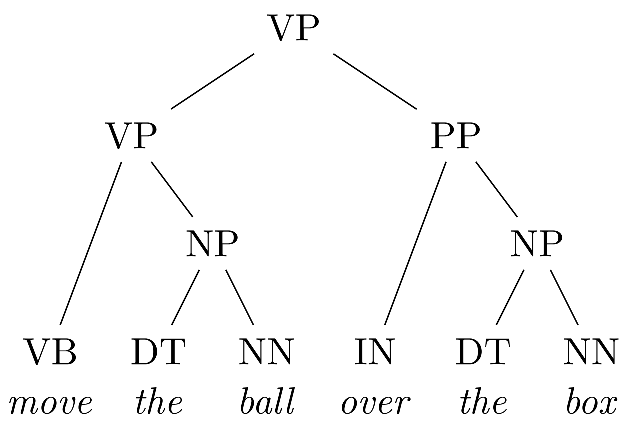


Figure 2. Parse tree for the instruction "move the ball over the box".

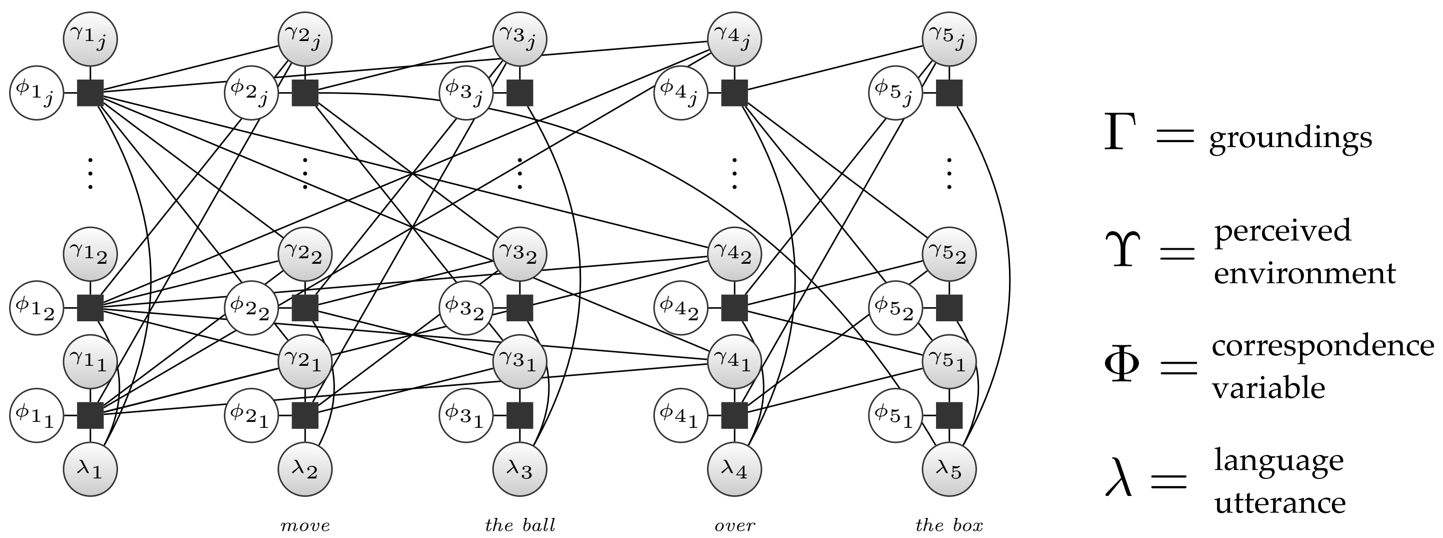


Figure 3. DCG for the instruction "move the ball over the box".

To perform efficient search, the DCG model incorporates observations regarding the conditional independence of both linguistic and grounding constituents, allowing the factored form in equation (2). Note that the prior robotic behavior is incorporated after the root phrase evaluation and therefore is not a parameter to the factor evaluation.

$$\arg\max_{\phi_{ij} \in \Phi} \prod_i \prod_j f\left(\phi_{ij}, \gamma_{ij}, \lambda_i, \Gamma_{c_{ij}}, \Upsilon\right) \qquad (2)$$

$$f\left(\phi, \gamma, \lambda, \Gamma_c, \Upsilon\right) = \frac{e^{\sum_k w_k f_k(\phi,\gamma,\lambda,\Gamma_c,\Upsilon)}}{\sum_{\phi_m \in \Phi} e^{\sum_k w_k f_k(\phi_m,\gamma,\lambda,\Gamma_c,\Upsilon)}} \qquad (3)$$

Results

We implemented an end-to-end system allowing a pilot study with non-expert users. Each experimental trial used a distinct set of initial commands, and the user described a correction via unrestricted language. Of 21 full trials, 8 resulted in a failed parse, 3 inferred an incorrect correction, and 10 correctly inferred and updated the trajectory.

Discussion

The size of our corpus was a limitation; in many of the incorrectly inferred cases, the parse failed because it occasionally encountered unknown words. This can be addressed via the incorporation of new training examples producing a more diverse lexicon.

Several instances of negative language occurred during data collection in which the user specified a correction while discouraging the prior behavior (e.g. "move the apple over the box **instead of** around it"). Our current model implementation is unable to understand such instructions because negation does not exist within our symbol space.

Future Work

We are currently pursuing the comprehension of negative robotic instructions. One approach involves the introduction of a novel symbol negation operator that would manipulate the space of symbols within the model to negate their expression. However, a naive implementation would double the search space of our model, thereby increasing run-time; given the real-time requirements, we intend to also pursue the incorporation of other DCG variants designed for improved search efficiency [1].

References

[1] O. Propp, I. Chung, M.R. Walter, and T.M. Howard. "On the Performance of Hierarchical Distributed Correspondence Graphs for Efficiency Symbol Grounding of Robot Instructions". In: Proceedings of the 2015 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems. Oct. 2015.

[2] T.M. Howard, S. Tellex, and N. Roy. "A Natural Language Planner Interface for Mobile Manipulators". In: Proceedings of the Int. Conf. on Robotics and Automation (ICRA). IEEE, May 2014, pp. 6652– 6659.

Presented at the 2016 Midwest Robotics Workshop