Natural Language Commands to a Robot Control System Automatically

Ayesha Akter,ID-21166014 Brac University,CSE Department

Abstract

Actions and control structures that can be readily implemented by natural language commands to a robot execution system automatically. I have a plan to learn a parser based on example like English commands and corresponding control language expressions. The technical nature of my formal representation allows a robot to interpret path instructions online while moving through a previously unknown environment.

1 Introduction

In this paper, I want to work on grounding natural language–interpreting human language into semantically informed structures in the context of robotic perception and actuation. Natural language (NL) is a rich intuitive mechanism by which humans can interact with systems around them, offering sufficient signal to support robot task planning. Human path instructions include complex language constructs, which robots must be able to execute without being given a fully specified world model. My goal is to investigate whether it is possible to learn a parser that produces correct, robot-executable commands for such instructions.

2 Related Work

Navigation of Robot is a critical and widely-studied task in mobile robotics, and following natural-language instructions is a key component of natural, multi-modal human or robot interaction. Parsing natural language to expressive formal representations such as [1]lamda-calculus has been demonstrated. Lamda-calculus is able to represent complex robot control systems. However, the unsurpassed of my knowledge such on Natural Language Commands to a Robot Control System Automatically- parser learning approaches have been applied in the context of robotics but I want to do on my approach in fig-1.

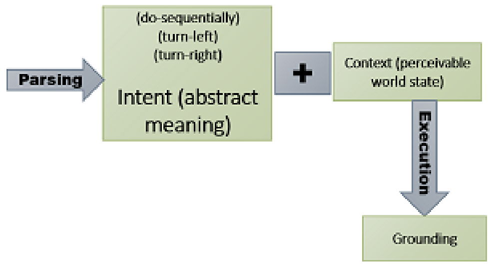

Figure 1: The task: Going from NL to robot control. First, the natural language command is parsed into a formal, procedural description representing the intent of the person. The robot control commands are then used by the executor, along with the local state of the world, to control the robot, thereby grounding the NL commands into actions while exploring the environment.

3 Technical Approach

My approach is to parse natural language into a high-level specification of robot behavior, which can then be executed in the context of an arbitrary environment. My approach also separates the parsing problem from execution, providing a more appropriate layer of abstraction for robot control. This architecture is shown in Fig. 2.

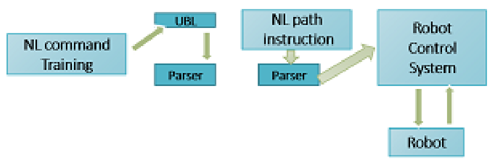

Figure 2: The high-level architecture of the end-to-end system. Training is performed by learning a parser from English instructions to RCL. In the experimental phase, the learned parser maps NL instructions to an RCL program that is executed by the robot.

I model control structures in a logic-based Robot

Control Language (RCL), inspired by task execution systems such as PRS [7], RPL [2], and GOLEX [6]. For a given set of route instructions, RCL represents the high-level execution intended by the person. Fig. 3 illustrates RCL programs for paths through an automatically generated, semantically labeled map [5], along with the instructions corresponding to that path. As such, RCL expressions can encode actions with nontrivial subtasks such as combined perception and action. For example, (exists left-loc) is a high-level command to the execution system to determine whether there is an opening to the left, while (move-to forward-loc) tells the robot to move one node forward in the semantic map.

locations	
current-loc:loc	robot's current position
forward-loc:loc	location ahead of robot
left-loc:loc	to robot's left
right-loc:loc	to robot's right
exists:t [loc]	does [loc] exist?

```
"Go left to the end of the hall."
(do-sequentially
( turn-left
(do-until
(or
(not
(exists forward-loc))
(room forward-loc))
(move-to forward-loc)))
```

Figure 3: The complete list of terms in Robot Control Language.

This also allows the language to be quite compact; a full list of RCL commands is given in Fig. 3(a), including terms for logical expressions, queries about local state, and actions to be performed. RCL is a formal language defined by a grammar; parsing is the process of producing an expression in that grammar from some input. In this case, the input is natural language route instructions, and the parsing target is a statement in RCL that can be passed to a robot controller for planning or further disambiguation.

Importantly, Unification Based Learner(UBL) can learn a parser solely from training data of the form f(xi; zi)g, where xi is a natural-language sentence and zi is a corresponding semantic language sentence. In brief, UBL learns a model for p(zi;yijxi;q), where q parameterizes the learned grammar G and yi is a derivation in G (zi is completely specified by yi). UBL uses a log-linear model: In [8], the lexicon—the set of lexical en-

$$p(z_i, y_i | x_i; \theta) \propto e^{\theta \cdot \phi(x_i, y_i, z_i)}$$

tries and their weights—was initialized with entries covering the entirety of each training example: for each pair of terms found in (xi; zi), one initial lexical entry was created. In this work I want to use a new initialization that is better able to handle our semantic language. Intuitively, rather than generating lexical items from each word, I want to allow arbitrary subsequences of natural-language words, along with semantic subexpressions as defined by the splitting procedure of [8]. As before, this favors (NL, semantics) pairs that are very predictive of each other, while remaining tractable.

4 Dataset Maps

Language datasets were generated as follows. First, English training and testing data from earlier work on paper [10] was re-annotated in RCL; as in [10] and [4], all English instructions are segmented into individual movement phrases.

5 Experiments

Experimentally, I am interested in whether the robot reaches the desired destination by exactly following the described path-baseline, parser and route-following experiment.

6 Conclusion

It is possible to learn a parser able to handle complex, procedural natural language commands for robot instruction automatically. This makes it possible to target a rich robot control language capable of handling abstract notions such as counting, loops, and conditionals. As an example, consider the different, contextually appropriate interpretations of the word 'go' where the system learned to interpret 'go to' and 'go left' as having quite different meanings. I want to establish that the help of parsing into a procedural language does require extensions to the parser learning process.

7 References

1. Y. Artzi and L.S. Zettlemoyer. Bootstrapping semantic parsers from conversations. In Proc. of the Conf. on Empirical Methods in Natural Language Processing, 2011.

2. M. Beetz, T. Arbuckle, T. Belker, M. Bennewitz, W. Burgard, A.B. Cremers, D. Fox, H. Grosskreutz, D. H¨ahnel, and D. Schulz. Integrated plan-based control of autonomous service robots in human environments. IEEE Intelligent Systems, 16(5), 2001.

3. J. Dzifcak, M. Scheutz, C. Baral, and P. Schermerhorn. What to do and how to do it: Translating natural language directives into temporal and dynamic logic representation for goal management

and action execution. In Proc. of the 2009 IEEE Int'l Conf. on Robotics and Automation (ICRA '09), 2009.

4. D.L. Chen and R. Mooney. Learning to interpret natural language navigation instructions from observations. In Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI-2011), pages 859–865, 2011.

5. S. Friedman, H. Pasula, and D. Fox. Voronoi random fields: Extracting topological structure of indoor environments via place labeling. In IJCAI, pages 2109–2114, 2007.

6. D. H¨ahnel, W. Burgard, and G. Lakemeyer. GOLEX - bridging the gap between logic (GOLOG) and a real robot. In Proc. of the German Conference on Artificial Intelligence (KI), Germany, 1998.

7. F. Ingrand, R. Chatila, R. Alami, and F. Robert. PRS: A high level supervision and control language for autonomous mobile robots. In Proc. of the IEEE International Conference on Robotics Automation (ICRA), 1996. 8. T. Kwiatkowski, L.S. Zettlemoyer, S. Goldwater, and M. Steedman. Inducing probabilistic CCG grammars from logical form with higher-order unification. In Proc. of the Conf. on Empirical Methods in Natural Language Processing, 2010.

9. T. Kwiatkowski, L.S. Zettlemoyer, S. Goldwater, and M. Steedman. Lexical generalization inCCG grammar induction for semantic parsing. In Proc. of the Conf. on Empirical Methods in Natural Language Processing, 2011.

10. C. Matuszek, D. Fox, and K. Koscher. Following directions using statistical machine translation. In ACM/IEEE International Conference on Human-Robot Interaction (HRI), 2010.

11. L.S. Zettlemoyer and M. Collins. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In Proc. of the Conf. on Uncertainty in Artificial Intelligence, 2005.