Workshop on Multimodal Semantics for Robotic Systems (MuSRobS)
IEEE/RSJ International Conference on Intelligent Robots and Systems 2015

Connecting natural language to task demonstrations and low-level control of industrial robots

Maj Stenmark, Jacek Malec
Dept. of Computer Science, Lund University, Sweden

Abstract— **Industrial robotics is a complex domain, not easily amenable to formalization using semantic technologies. It involves such disparate aspects of the real world as geometry, dynamics, constraint-satisfaction, planning and scheduling, real-time control, robot-robot and human-robot communication and, finally, intentions of the robot user. To represent so different kinds of knowledge is a challenge and the research on combining those topics is only in its infancy.**

This paper describes our attempts to combine descriptions of robot tasks using natural language together with their realizations using robot hardware involving force sensing, ultimately leading to a potential of learning new robot skills employing force-based assembly. We believe it is a novel approach opening possibilities of semantic anchoring for learning from demonstration.

I. Introduction

Recent developments in robotics, artificial intelligence and cognitive science lead to bold predictions about the soon-to-come robotization of all aspects of human life. Robots will help the elderly, perform mundane jobs no one wants, drive our cars, fill our refrigerators when needed, tirelessly rehabilitate patients in need of physical exercise, fight our wars, become our sex partners, etc., etc. Some draw even the conclusion that robots will take over Earth and will turn humans into obsolete pets.

However, when observing the development of the robotics field, we can realize that this perspective is rather far, far away. Service robots are clumsy and unskilled, no one trusts a robotized car, and production still relies on simple manipulators programmed in a classical manner by skilled engineers. Any attempt to instruct a robot to perform a concrete manufacturing task consists of person-weeks of work of skilled system integrator engineers, who take into account the geometrical layout of the workcell, all the objects involved, involving their geometry, physical properties and, last but not least, purpose of the task. It is the implicit knowledge that needs to be transferred into robot code, which makes this task so complex.

Use of semantic technologies is advocated since at least a decade. Unfortunately, industrial robotics is a complex domain, not easily amenable to formalization. It involves such disparate aspects of the real world as geometry, dynamics (including forceful interaction with the work objects), constraints, planning, scheduling, optimization, real-time control, robot-robot and human-robot communication and, finally, intentions of the robot user. To represent so different kinds of knowledge is a challenge and the research on combining those topics is only in its infancy.

In particular, we have devoted last years to understand and describe robotic assembly, including force-based operations (snap, drill, press, etc.), using machine-readable formalism expressing the semantics of possible robot actions. Without it there will be no possibility to create a meaningful reasoning leading from the task specification (what needs to be manufactured) to task synthesis (how can this be achieved using the available robot skills) and robust execution by the synthesized code for a particular architecture. Not to mention swift error handling in case of unexpected problems, portability of a robot skill from one robot to another, and learning of new skills.

In previous research, we have focused our attention on two areas: interaction between the user and the robotic system, preferably on the user's conditions, e.g., using natural language [1], [2], and representation of force-controlled assembly operations, particularly problematic due to the inherent mix of continuous and discrete aspects [3]. Besides being able to talk with the robot about a force-controlled assembly operation, we would like it to be learnt automatically from a demonstration and be represented semantically in a manner enabling portability among different robots.

So far these kinds of systems are developed only in research laboratories. Our own research is done in the context of several EU projects, in particular ROSETTA, PRACE and SMEROBOTICS, aiming at developing intelligent interactive systems suitable for inexperienced users, such as SMEs. Before they reach the factory floor though, they need to be filled with sufficient production knowledge so that they become useful. Knowledge acquisition is a bottleneck in developing practical systems, as it can only happen while the system is used, but it won't be useful before it is done: a classical chicken and egg problem. Therefore the only viable solution is a learning system, capable of sharing its experiences by storing them in a (possibly cloud-based) knowledge base [4] and using experiences of other robots by importing and adapting their skills. However, such a solution requires a common understanding of the contents of this knowledge base, thus, a commonly agreed-upon semantics.

The work on standardization of robotics domain is already quite well advanced. There exist ontologies for specific domains, like service robotics or surgical robotics, and a core ontology for robotics and automation (CORA) recently standardized by IEEE [5]. However, they introduce concepts in symbolic form without properly connecting to all their denotations, e.g., robot programs instantiating skills named in these ontologies. Our work addresses this problem by pro-

viding concrete denotations belonging to several modalities. As we have mentioned before, we describe robot actions using natural language, using assembly graphs, using transition systems, using iTaSC formalism, and using the actual robot code. Those multiple modalities are co-existing in one system, letting the reasoner switch between representations when such a need arises.

Learning from demonstration leads to new problems in semantic anchoring of robot actions, as there is no obvious, apparent meaning in robot movements. Semantics may be either guessed, derived by inductive reasoning, or attributed post factum by humans via some form of annotating. In particular, force-based assembly is problematic as quite often the difference between success and failure depends on a particular profile of the force signal. So far, this issue has been approached using sensor fusion techniques, without direct support from semantics. Our work attempts to remedy this situation, introducing natural language into the picture and letting assembly to be not only detected via sensor readings, but also by being simultaneously told about.

II. Related work

In the domain of service robotics, there are some interesting frameworks for representation of household tasks and environments. KnowRob [6] is a knowledge processing system that combines declarative and procedural knowledge from multiple sources, e.g., the RoboEarth [7] database and web sites. A similar project is RoboHow [8], which developed a knowledge-based reasoning service OpenEASE [9] and attempts at bridging the gap from symbolic planning to constraint-based control [10]. Ontologies for kit building applications for industrial robots have been developed by Balakirsky et al. [11] and Carbonera et al. [12] developed a ontology for positions. We have already mentioned the standardization work of IEEE Working Group ORA [5].

We are interested in integrating low-level statistical task representations taken from demonstrations. Such tasks can be represented by a trajectory or force profile. The trajectories can be extracted from the demonstration by first applying segmentation algorithms and then parameterizing each segment as a trajectory. Niekum et al. [13], [14] use Beta Process Autoregressive Hidden Markov Models from Fox et al. [15] to automatically segment demonstrations and dynamic movement primitives (DMPs) [16] to represent the trajectories. Since the statistical properties of semantically different sub-tasks can be similar, they use predecessor states to refine the classification and determine the transitions in a finite state machine. Other learning methods are for example reinforcement learning, used by Metzen et al. [17] to learn skill templates, and Iterative Learning Control, used by Nemec et al. [18] to follow demonstrated force profiles.

One way to annotate objects and actions is to describe them using natural language. Matuszek et al. [19], Kollar et al. [20] and Landsiedel [21] use natural language to describe routes and Walter et al. [22] use language descriptions to semantically annotate maps. She et al. [23] studies the dialogue system, while Cakmak [24] evaluates methods for teaching operators how to interact with a robot using kinesthetic teaching and dialogue.

Please note that our understanding of the term *multimodal semantics* differs from the one quite commonly encountered in literature, see e.g. [25], where the authors aim at finding the meaning of a particular text fragment using the statistical approach grounded both in text and image corpora. However, there is no attempt to use this semantics in the reverse direction, to generate new utterances (that our robot programs would correspond to).

III. Current work

The focus of current work is to semantically annotate task demonstrations to enable reuse and reasoning. This involves annotating log data with quantities, units, and task states. The logs can then be used to identify force/torque and position constraints and application specific parameter values (positions, velocity, stiffness, etc.). The demonstrations are use to segment the task in different sub-skills and extract parameters for each skill. One approach is to describe the trajectory of the sub-skills using DMPs and then parameterize the primitives and describe them with for example skill type, preconditions, and postconditions.

As an example, when demonstrating picking and placing of an object, the task can be segmented into different sub-skills. First the robot approaches the object, opens the gripper, moves into pick position, closes the gripper, retracts from the surface moves to the place position (perhaps using via positions as well), positions the object correctly, releases it and retracts. Each segment can be described using a trajectory (e.g., a DMP) in some reference frame together with a gripper state. Multiple demonstrations can be used for each sub-skill in order to detect which for example relevant reference frame and allowed gripping poses. To enable reuse, we are working on annotating the segments with initial allowed start positions and gripper state, a skill type and postconditions. This will allow the planner to add required actions before or after the skill and add error-handling procedures to the task (e.g., if the robot drops the object when transporting it, the object should be localized and picked up again). The skill also has to be parameterized so that it can be initialized correctly, for example, specifying controller, reference frames, velocity values.

Another example is force-controlled assembly. The force data is not used for sensor fusion, it is used to control the motions of the robot and signals failure or success of the assembly. In a snapfit assembly skill, where a two plastic pieces, a switch and a box shown in Fig. 1, are ”snapped” together, the force signature indicates whether the snap occurred or not. In previous work [1] such task could be expressed using the force constraint directly in *g*uarded motions. Using a graphical user interface, primitive actions and skills could be combined into a sequence. An example is shown in Fig. 2. In the sequence, the box is first picked and placed on a fixture using three search motions. The first motion moves the robot down until it feels contact forces in the z-direction, then, while pressing down, it searches in

the y-direction until contact with the wall, finally, it searches in the x-direction while simultaneously pressing down and towards the wall. In the sequence `pickbox`, `movetofixt`, `pickswitch` and `retract` are position based motions running on the native robot controller. The `snapFitSkill` is a reused skill, which in turn contains multiple guarded searches. From the graphical representation, the skill specification can be exported to XML-format (see excerpt in Fig. 3) and to runnable format, see Fig. 4. The skills are semantically annotated with sensor and controller type, and the parameters are described with units.

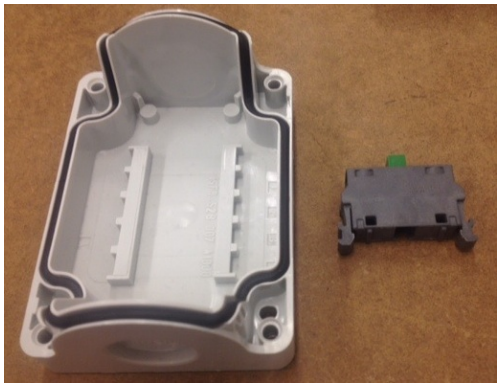

Fig. 1. A part of a box to the left and a switch to the right.

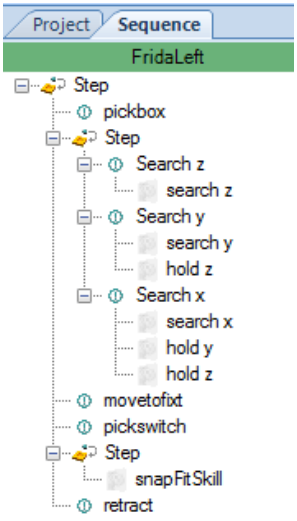

Fig. 2. An example of the user interface.

In another assembly, a rectangular metal plate (a shield can) is inserted on a printed circuit board (PCB). The PCB is attached in a fixture, which is attached to a force sensor. The assembly starts by tilting the shield can above the PCB (see Fig. 5), moving down until the corner touches the board. Then, the robot attaches one corner of the plate to a corner on the PCB and rotates the plate into place. The rotation is first carried out around the xy-plane of the PCB until either the long or the short side of the rectangle touches the PCB, then the last side has to be rotated into place. That is, if the

```
<SkillSpecification>
  <Frame id="f1">
    <origin>[ 490 , 6 , 43 ]</origin>
        <quaternion>[ 1 , 0 , 0 , 0]</quaternion>
  </Frame>
  <ToolTransform id="tool1">
    <trans>[0,0,87]</trans>
        <quaternion>[0,-0.707106781,0.707106781,0]</quaternion>
  </ToolTransform>
  <ImpedanceControlParams id="z-controller">
    <M>0.01</M>
    <D>0.2</D>
  </ImpedanceControlParams>
  <ImpedanceControlParams id="y-controller">
    <M>0.02</M>
    <D>0.6</D>
  </ImpedanceControlParams>
  <Action id="z-search" tool="tool1">
    <Direction>
      <searchVelocity unit="mm/s">-30</searchVelocity>
      <motionframe>f1</motionframe>
      <motiondir>z</motiondir>
      <threshold unit="N">3</threshold>
    </Direction>
  </Action>
  <Action id="y-search" tool="tool1">
    <Direction>
      <searchVelocity unit="mm/s">-40</searchVelocity>
      <motionframe>f1</motionframe>
      <motiondir>y</motiondir>
      <threshold unit="N">3</threshold>
    </Direction>
    <Constraint>
      <type>forcecontrolled</type>
      <controllerId>z-controller</controllerId>
      <motionframe>f1</motionframe>
      <motiondir>z</motiondir>
      <value unit="N">3</value>
    </Constraint>
  </Action>
```

Fig. 3. The XML representation of the three guarded search motions created in the GUI.

longer side of the rectangle is parallel with the x-axis and the rotation around a xy-vector from the initial tilted position will align it with the PCB, the execution will branch into a rotation around the x-axis until the short side is aligned with the PCB. Otherwise, the rotation will align the short side first, as seen in Fig. 6.

To lower the threshold for the user, we want to use natural language dialogues to describe the demonstration and extend the task. Together with the parameterized demonstrations, this will allow the user to use high-level structures such as loops and if-then-else statements, which are easily described using language but tedious or difficult to describe using demonstrations only. In our current system, the user can instruct the robot using unstructured text or dictate the task using Google dictation tools. An example instruction is displayed in Fig. 7. All parameters have default values, which makes the high-level nominal task easy and fast to generate from text. The programming interface use language specific statistical tools to extract the semantics of the sentences, then a rule-based mapping to robot skills and world objects. At the moment, English and Swedish [26] are supported programming languages.

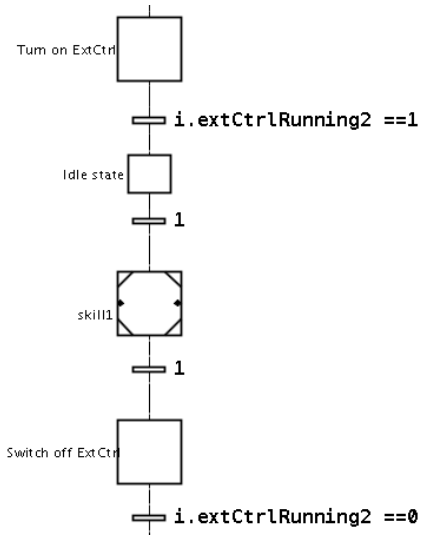

Fig. 4. Part of the executable state machine. The guarded searches run on an external force controller (ExtCtrl) which has to be turned on and off before the force controlled skills are executed.

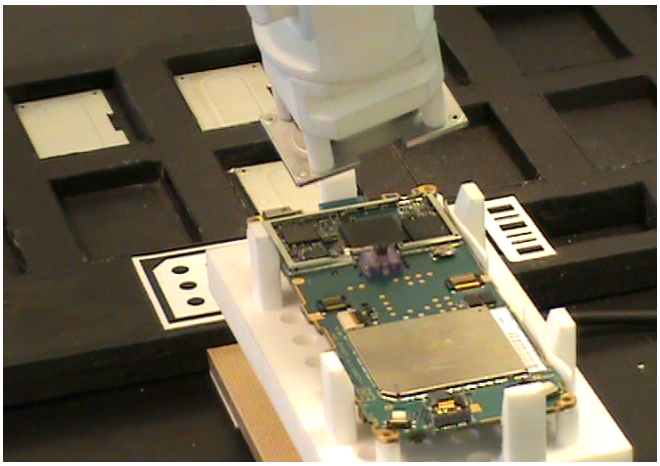

Fig. 5. In the initial position, the shield can is tilted above the PCB.

IV. Conclusions

The immediate future work involves investigating how to teach pre-and postconditions for skills learned from demonstration, to enable online reasoning. These conditions need to be anchored in sensor readings. Inductive inference is one possibility; another is to use mixed-initiative dialogue with the user, asking for guidance or confirmation, yet another is to introduce some annotation tool to be used simultaneously with the learning procedure.

It is desirable to have natural language support on all levels in the system. At the moment, we only support task instruction, but we also want to be able to describe the world and connect the perceived objects and situations to (new) semantic symbols. E.g., saying ”This is a nut” after teaching the camera system to recognize an object, or describing a pallet as ”empty”. At the moment, the robot is a passive participant in the dialogue, only reacting on commands from the human. When interacting with non-expert users, the robot

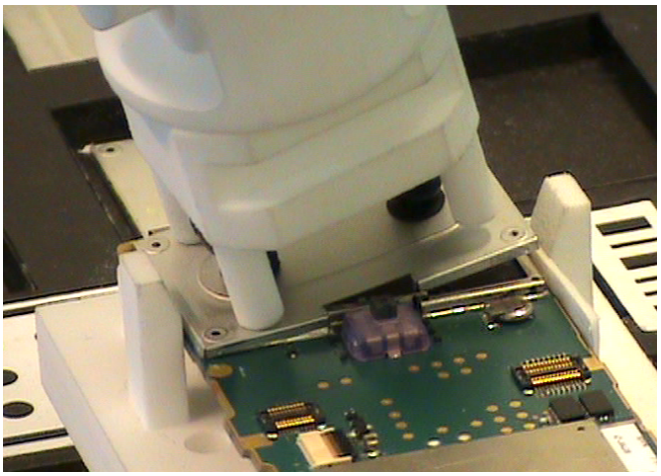

Fig. 6. The short side is aligned with the PCB.

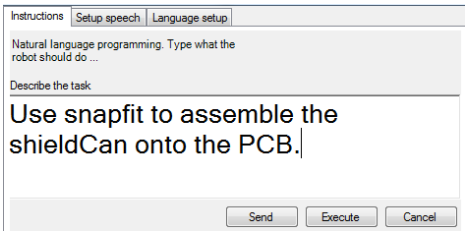

Fig. 7. The user can instruct the robot using unstructured natural language commands.

should ask questions and come with suggestions on what to do.

The next step is to introduce the possibility of extending the robot knowledge by adding new concepts to the semantic hierarchy. This is a more complex task than the previous one, as it involves inducing relations with existing concepts and proper placing of the new symbol in the IsA hierarchy.

Yet another interesting problem is to reason about “synonyms” among robot programs, i.e. syntactically different structures or programs leading to the same effect. A simple example is a “localize and pick” task that may use different kinds of sensors to localize an object, while the goal (of picking the object from its current location) is achieved irrespectively of which concrete sensor is used. How to teach the system that two skills are equivalent in such (or some other) sense? What needs to be told? What kind of reasoning performed?

Representing knowledge about industrial processes involving semantically-capable robots is a challenge leading to fascinating questions. We are quite sure we will have a lot to do in years to come.

Acknowledgments

The research leading to these results has received partial funding from the European Union’s seventh framework program under grant agreement No. 287787 (project SME-robotics) and from the European Union’s H2020 program under grant agreement No. 644938 (project SARAFun).

REFERENCES

[1] M. Stenmark, "Instructing industrial robots using high-level task descriptions," Ph.D. dissertation, Lund University, Department of Computer Science, Mar. 2015, licentiate Thesis.

[2] M. Stenmark and J. Malec, "Knowledge-Based Instruction of Manipulation Tasks for Industrial Robotics," *Robotics and Computer Integrated Manufacturing*, vol. 33, pp. 56–67, 2015. [Online]. Available: http://lup.lub.lu.se/record/4679243/file/4679245.pdf

[3] J. Malec, K. Nilsson, and H. Bruyninckx, "Describing assembly tasks in declarative way," in *Proc. IEEE ICRA 2013 Workshop on Semantics, Identification and Control of Robot-Human-Environment Interaction*, Karlsruhe, Germany, May 2013, pp. 50–53.

[4] M. Stenmark, J. Malec, K. Nilsson, and A. Robertsson, "On Distributed Knowledge Bases for Robotized Small-Batch Assembly," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 2, pp. 519–528, 2015. [Online]. Available: http://dx.doi.org/10.1109/TASE.2015.2408264

[5] "IEEE standard ontologies for robotics and automation," IEEE Standard 1872-2015, 2015.

[6] M. Tenorth and M. Beetz, "Knowrob: A knowledge processing infrastructure for cognition-enabled robots," *The International Journal of Robotics Research*, vol. 32, no. 5, pp. 566–590, 2013.

[7] M. Tenorth, A. Perzylo, R. Lafrenz, and M. Beetz, "Representation and exchange of knowledge about actions, objects, and environments in the roboearth framework," *Automation Science and Engineering, IEEE Transactions on*, vol. 10, no. 3, pp. 643–651, July 2013.

[8] M. Tenorth, G. Bartels, and M. Beetz, "Knowledge-based specification of robot motions," in *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, 2014.

[9] M. Beetz, M. Tenorth, and J. Winkler, "Open-EASE – a knowledge processing service for robots and robotics/ai researchers," in *IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, Washington, USA, 2015.

[10] E. Scioni, G. Borghesan, H. Bruyninckx, and M. Bonfe, "Bridging the gap between discrete symbolic planning and optimization-based robot control," in *2015 IEEE International Conference on Robotics and Automation*, 2015.

[11] S. Balakirsky, Z. Kootbally, C. Schlenoff, T. Kramer, and S. Gupta, "An industrial robotic knowledge representation for kit building applications," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, Oct 2012, pp. 1365–1370.

[12] J. Carbonera, S. Rama Fiorini, E. Prestes, V. Jorge, M. Abel, R. Madhavan, A. Locoro, P. Goncalves, T. Haidegger, M. Barreto, and C. Schlenoff, "Defining positioning in a core ontology for robotics," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, November 2013, pp. 1867–1872.

[13] S. Niekum, S. Chitta, A. Barto, B. Marthi, and S. Osentoski, "Incremental semantically grounded learning from demonstration," Berlin, Germany, 2013.

[14] S. Niekum, S. Osentoski, G. Konidaris, S. Chitta, B. Marthi, and A. G. Barto, "Learning grounded finite-state representations from unstructured demonstrations," *The International Journal of Robotics Research*, vol. 34, no. 2, pp. 131–157, 2015.

[15] E. B. Fox, M. I. Jordan, E. B. Sudderth, and A. S. Willsky, "Sharing features among dynamical systems with beta processes," in *Advances in Neural Information Processing Systems 22*, Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, Eds. Curran Associates, Inc., 2009, pp. 549–557.

[16] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Learning attractor landscapes for learning motor primitives," in *Advances in Neural Information Processing Systems 15 (NIPS2002)*, 2002, pp. 1547–1554.

[17] J. Metzen, A. Fabisch, L. Senger, J. de Gea Fernndez, and E. Kirchner, "Towards learning of generic skills for robotic manipulation," *KI - Knstliche Intelligenz*, vol. 28, no. 1, pp. 15–20, 2014.

[18] B. Nemec, F. Abu-Dakka, B. Ridge, A. Ude, J. Jorgensen, T. Savarimuthu, J. Jouffroy, H. Petersen, and N. Kruger, "Transfer of assembly operations to new workpiece poses by adaptation to the desired force profile," in *Advanced Robotics (ICAR), 2013 16th International Conference on*, Nov 2013, pp. 1–7.

[19] C. Matuszek, E. Herbst, L. Zettlemoyer, and D. Fox, "Learning to parse natural language commands to a robot control system," in *Experimental Robotics*, ser. Springer Tracts in Advanced Robotics. Springer International Publishing, 2013, vol. 88, pp. 403–415.

[20] T. Kollar, S. Tellex, D. Roy, and N. Roy, "Grounding verbs of motion in natural language commands to robots," in *Experimental Robotics*, ser. Springer Tracts in Advanced Robotics. Springer Berlin Heidelberg, 2014, vol. 79, pp. 31–47.

[21] C. Landsiedel, R. de Nijs, K. Kuhnlenz, D. Wollherr, and M. Buss, "Route description interpretation on automatically labeled robot maps," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, May 2013, p. 22512256.

[22] M. R. Walter, S. Hemachandra, B. Homberg, S. Tellex, and S. Teller, "Learning semantic maps from natural language descriptions," in *Proceedings of the 2013 Robotics: Science and Systems IX Conference*, Berlin, Germany, 2013.

[23] L. She, S. Yang, Y. Cheng, Y. Jia, J. Chai, and N. Xi, "Back to the blocks world: Learning new actions through situated human-robot dialogue," in *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*. Philadelphia, PA, U.S.A: for Computational Linguistics, 2014, pp. 89–97.

[24] M. Cakmak and L. Takayama, "Teaching people how to teach robots: The effect of instructional materials and dialog design," in *International Conference on Human-Robot Interaction (HRI)*, Bielefeld, Germany, Mar. 2014.

[25] E. Bruni, N. K. Tran, and M. Baroni, "Multimodal distributional semantics," *J. Artif. Int. Res.*, vol. 49, no. 1, pp. 1–47, Jan. 2014.

[26] M. Stenmark, "Bilingual robots: Extracting robot program statements from swedish natural language instructions," in *Proc. of The 13th Scandinavian Conf. on Artificial Intelligence*, Halmstad, Sweden, 2015.