Efficient Natural Language Interfaces for Assistive Robots

Thomas M. Howard[1], Istvan Chung[2], Oron Propp[3], Matthew R. Walter[1], and Nicholas Roy[1]

***Abstract*— Language is a powerful tool that enables humans and robots to interact without the need for complex graphical interfaces. Statistical techniques for interpreting the meaning of utterances in complex domestic environments, however, remain computationally intensive and are prone to error. Herein we present a model for language understanding that uses parse trees and environment models to infer both the structure and the meaning of probabilistic graphical models for symbol grounding. This model, called the Hierarchical Distributed Correspondence Graph (HDCG), exploits information about symbols that are expressed in the corpus to learn rules that describe how to construct graphical models that are faster to search. In a comparative experiment, we observe an order of magnitude improvement in the speed of probabilistic inference over the Distributed Correspondence Graph (DCG) model. We conclude with a discussion of potential applications in rehabilitation and assistive robotics and future directions of research.**

I. INTRODUCTION

Ideas from robotics have the potential to improve the quality of life for people with physical, cognitive, and visual impairments by allowing them to navigate in, interact with, and perceive their environment independent of human aids. Recent investigations into networks for patient tracking [1] and robot guided therapy [2], [3] show that we can improve the efficiency and effectiveness of patient care through the utilization of such technologies. Smart wheelchairs have been proposed to address limited mobility and have been an active area of research over the past three decades [4]. A barrier to widespread adoption of smart wheelchairs is the way in which we interact with such systems. Interfaces based on haptic feedback [5], electrooculography [6], electroencephalography [7], and recognition of facial gestures [8] have been explored as improvements over manual control. Natural language interfaces are another promising method for human-robot interaction and control of cyber-physical systems. Recognizing its importance, there has been an increased focus on developing a capacity for robots to understand natural language instructions in the context of route direction following [9]–[13], map building [14], and object manipulation [15], [16].

A key component of natural language interfaces is the model for providing physical meaning for linguistic constituents [17]. The Generalized Grounding Graph (G^3) model [15] exploits the hierarchical structure of language to construct a probabilistic graphical model composed of groundings ($\gamma \in \Gamma$), correspondences ($\phi \in \Phi$), phrases ($\lambda \in \Lambda$), and the physical environment (Ψ). The unknown groundings are searched for values that maximize the product of factors that represent a true correspondence to phrases and connected groundings. The Distributed Correspondence Graph (DCG) model [18] expands on this idea by assuming conditional independence of grounding constituents to improve the computational efficiency of probabilistic inference. This model searches the unknown correspondences to maximize the product of factors (f) that evaluate the correlation between phrases, groundings, and the physical environment:

$$\arg\max_{\Phi} p\left(\Phi|\Gamma,\Lambda,\Psi\right) = \arg\max_{\Phi} \prod_i \prod_j f\left(\Phi_{i_j},\Gamma_{i_j},\Lambda_i,\Psi\right) \quad (1)$$

Duvallet et al. [19] applies the DCG model to infer both environment maps and robot behaviors from natural language utterances to control a robotic wheelchair in unknown environments. Both of these models take a brute force approach to constructing the probabilistic graphical models, without considering whether or not a particular grounding, relationship, or correspondence should be included or excluded from the search. This has important consequences on the time required to the distribution of symbol groundings and subsequently limits the scalability of natural language interfaces to rudimentary tasks in simple environments.

In this paper, we explore the impact on computational efficiency of a hierarchy of graphical models that reasons over the structure of subsequent layers. Example applications in service robotics demonstrate that we can improve the scalability of algorithms for natural language understanding by learning how to construct concise probabilistic graphical models that are efficient to search with computational and

Fig. 1. An image of the initial state of a kitchen for a service robotics task. The physical environment consists of one sink, fourteen plates, five mugs, seven cups, one pitcher, one faucet, five spoons, six forks, four knives, one soap dispenser and one robot (not pictured) for a total of forty-six objects.

[1]T.M. Howard, M.R. Walter, and N. Roy are with the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

[2]I. Chung is with Buckingham Browne & Nichols School, Cambridge, MA 02138, USA

[3]O. Propp is with Gann Academy, Waltham, MA 02452, USA

temporal constraints. We conclude with a discussion of the potential impacts for natural language interfaces in service robotics.

II. HIERARCHICAL DISTRIBUTED CORRESPONDENCE GRAPHS

Consider the parse tree of the expression "pick up the mug on the plate to the left of the sink" that is illustrated in Figure 2 in the context of the environment depicted in Figure 1. When searching for the meaning of the expression with the DCG model, each of the fifteen phrases in the utterance must be searched for the set of correspondence variables that maximize the product of factors with all possible groundings from the physical environment. This means that the representation of the physical environment in the DCG model has a significant impact on the computational requirements of probabilistic inference. Commonly, environments have numerous objects, regions, and relationships that are irrelevant to the meaning of the expression and can be excluded from the search process. In the aforementioned example, we can interpret the need to consider certain relationships between a robot, a mug, a plate, and a sink, but cannot identify the specific robot, mug, plate, or sink among multiple objects of the same class without incorporating physical properties of the environment. Likewise, we can infer that the relationships between the robot and the soap dispenser, silverware, and the pitcher are irrelevant to the goal conditions of the request. For the environment in Figure 1, we may only need to consider twenty-one of the forty-six objects and a subset of the regions and constraints that involve those objects to understanding the meaning of the robot instruction. Efficient techniques for determining the correct resolution and representation of the physical world for understanding the meaning of an utterance remains an open question in human-robot interaction.

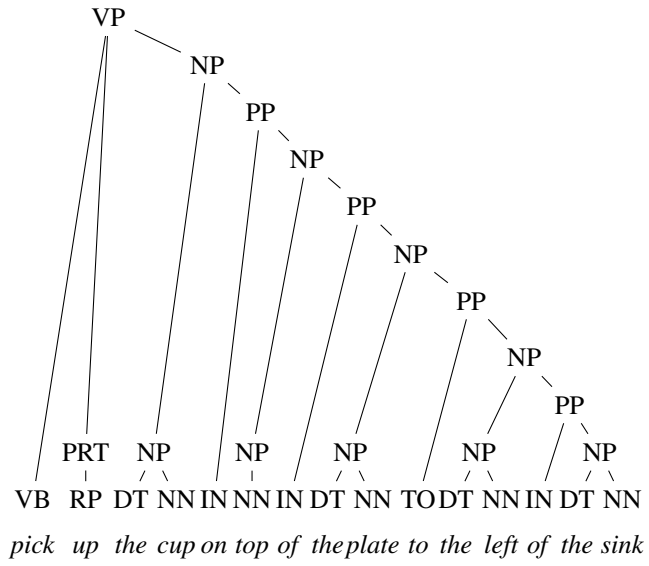

Fig. 2. The parse tree for the robot instruction "pick up the cup on top of the plate to the left of the sink".

Our approach to inferring the most likely representation of a search space for language understanding applies the DCG model hierarchically. At higher levels in the hierarchy, models are used to infer rules ($\mathtt{R}$) that are used to construct the space of groundings for subsequent models. At the base of the hierarchy, the space of groundings is constructed from the distribution of rules inferred by the previous layer.

$$\Gamma \rightarrow \Gamma(\mathtt{R}) \tag{2}$$

Figure 3 illustrates how rules are used to filter the space of region groundings for each phrase in the HDCG model. In this example, three object rules ($\mathtt{R}_1$, $\mathtt{R}_2$, and $\mathtt{R}_3$) and two region type rules ($\mathtt{R}_4$ and $\mathtt{R}_5$) are expressed to permit only region types LEFT and ABOVE and objects o_2, o_3, and o_5. These five rules permit the expression of six region groundings (γ_{11}, γ_{15}, γ_{19}, γ_{23}, γ_{35}, and γ_{39}) and eliminate 362 region groundings from consideration. Rules are also applied to filter object, relationship, and other types of groundings for each phrase in a similar manner.

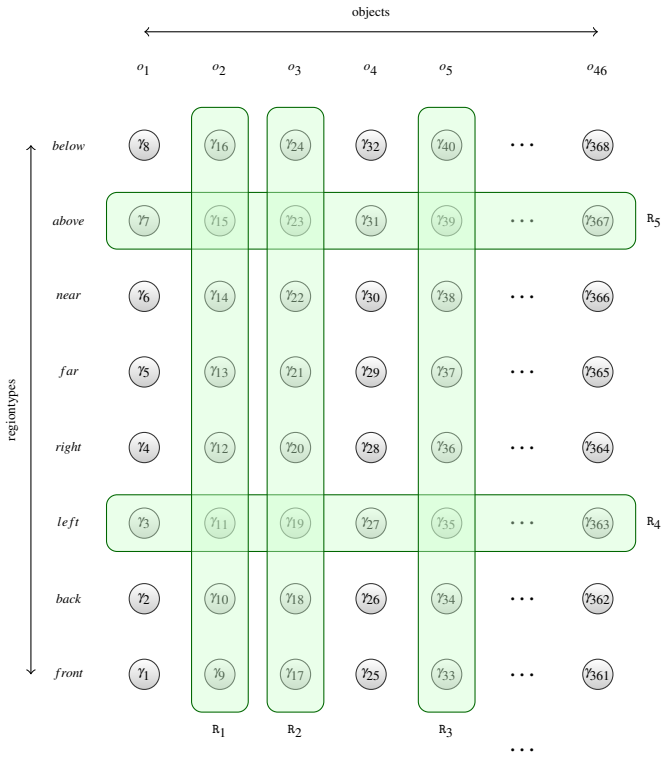

Fig. 3. An illustration of how rules are used to filter the space of region groundings. The space of regions groundings is formed by the intersection of rules that permit objects and region types. In this example, the intersection of three object rules and two region type rules allows six (γ_{11}, γ_{15}, γ_{19}, γ_{23}, γ_{35}, and γ_{39}) of 368 region groundings to be expressed.

Mathematically, we introduce latent variables that represent the rules for constructing the space of groundings that we integrate out to infer the most likely correspondence variables. We model the likelihood of correspondences given rules, groundings, language, and the physical world and the likelihood of rules given groundings, language, and the physical world as DCG models. The factors in each DCG model are represented as log-linear models that are trained

from a corpus of labeled examples.

$$\arg\max_{\Phi} \int_{\mathtt{R}} p\left(\Phi|\mathtt{R},\Gamma\left(\mathtt{R}\right),\Lambda,\Psi\right) p\left(\mathtt{R}|\Gamma\left(\mathtt{R}\right),\Lambda,\Psi\right) \tag{3}$$

$$\arg\max_{\Phi} \int_{\mathtt{R}} \prod_{i} \prod_{j} f\left(\Phi_{i_j},\Gamma_{i_j}\left(\mathtt{R}\right),\Lambda_i,\Psi,\mathtt{R}\right) \times \prod_{i} \prod_{j} f\left(\mathtt{R},\Lambda_i,\Psi,\Gamma_{i_j}\left(\mathtt{R}\right)\right) \tag{4}$$

Graphically, this model, which we call the Hierarchical Distributed Correspondence Graph (HDCG), can be represented as illustrated in Figure 4. Inference occurs from top to bottom, where a set of rules is inferred for constructing the space of groundings for the subsequent model. The lowest-level model contains the relationships in the physical environment that we want to infer from the natural language instruction. The search space of the highest-level model is fixed from the assumptions of how to partition the search spaces in subsequent models.

On the surface, it appears that we have made the model more computationally complex by adding additional factors and random variables for probabilistic inference. For infrequent expressions that require reasoning over relationships between all objects in the physical environment, this would be true. If the relationships between many objects in the scene are irrelevant to the desired task, however, it is beneficial to cull those random variables from the model that infers the relationships between physical objects. Returning to the original example, if we can remove considerations of regions to the right, below, front, and back of objects we can significantly reduce the number of relationships we need to evaluate. Likewise, if we are able to remove background objects that commonly appear in the kitchens (e.g., utensils, pots, pans), the hierarchical variation of the model can further eliminate many objects, regions, and relationships from consideration.

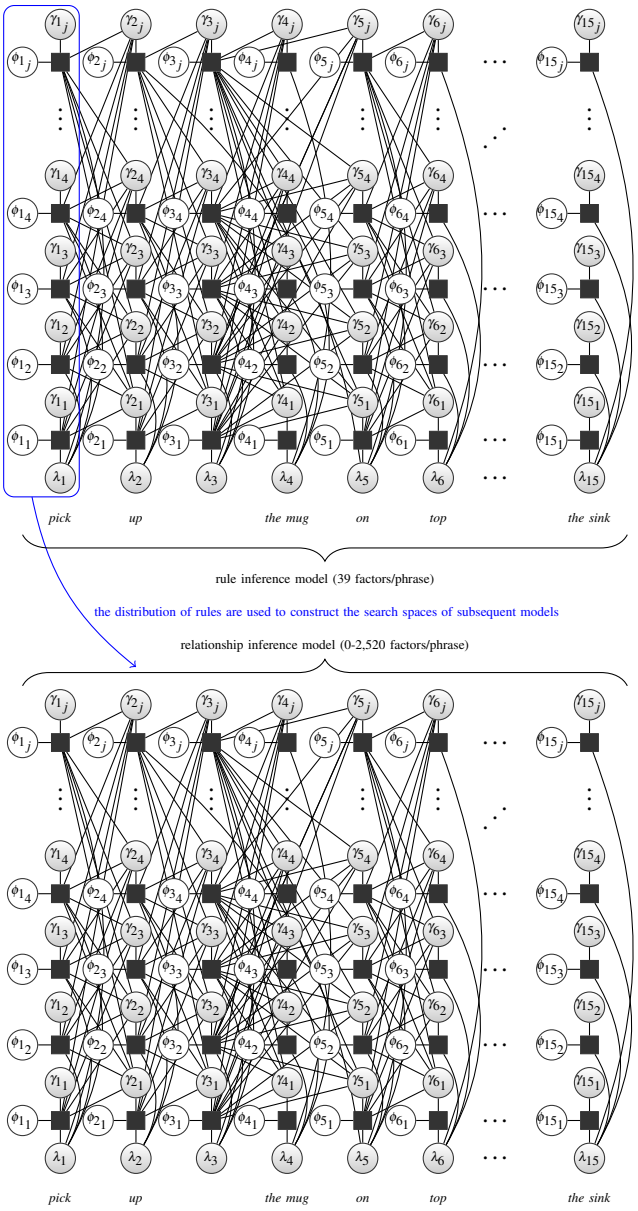

Fig. 4. An illustration of the Hierarchical Distributed Correspondence Graph composes of two models. A distribution of rules is inferred from the language and environment model that is subsequently used to construct models to infer the desired relationships between physical objects. The search space for each model in this figure applies the assumptions described in Section III for a world composed of four objects.

III. EXPERIMENTS

To evaluate the predicted improvement in computational efficiency, we compare the runtime of the DCG model against that of the HDCG model. Each model is trained and evaluated on the example corpus of the open-source H2SL software package,[1] which contains the meanings of thirty-two instructions, such as "walk to the chair and the crate" and "approach the front of the desk," in a symbolic language used to formulate problems for a trajectory planner. The version of H2SL that is used in this experiment assumes that objects are one of twenty-seven types, regions are one of eight types (FRONT, BACK, LEFT, RIGHT, ABOVE, BELOW, NEAR, or FAR), and constraints are one of two types (INSIDE or OUTSIDE). The space of groundings for each phrase is constructed from the possible objects, regions, and constraints that can be represented in a particular world. For example, each phrase in an instruction that assumes a world composed of four physical objects could represent four objects, thirty-two regions and/or over two thousand constraints. Inference is made efficient for the DCG model in H2SL by throttling the space of groundings for each type of phrase (e.g., constraints can only be groundings for verbs), though a more general approach would consider any possible constituent for a phrase grounding.

We assume that the HDCG model in this experiment is composed of two models: a rule inference model and a relationship inference model. We consider three types of rules that permit or exclude objects, regions, and constraints for rule inference in the HDCG model. The search space in the rule inference model is composed of rules parameterized

[1] https://code.google.com/p/h2sl/

by characteristics of the groundings, such as the type of object, region, or constraint. In this experiment, we assumed that there were twenty-seven object type rules, nine region type rules, and three constraint type rules. Since the space of rules is not a function of the environment, each phrase in the rule inference model has thirty-nine factors. Examples for training the rule inference model come from searching for the minimum set of rules that permit all expressed groundings in relationship inference examples. A set of rules is used to then partition the search space of the relationship inference model, whose constituents match those of the original DCG model. The log-linear models for rule and relationship inference used 5,244 and 3,316 features respectively.

IV. RESULTS

The experiment described in Section III was performed on a Apple MacBook Pro with a 2.6 GHz Intel Core i7 processor using a single-threaded C++ implementation of the DCG and HDCG models. Since the goal of this experiment was to compare the runtime performance of probabilistic inference, each model was trained and tested with each of the thirty-two examples. We observed no difference in the inferred symbols for both models in each of the thirty-two example instructions. A beam width of four is used to infer the meaning of each model and the top scoring set of constituents from the rule inference model is used to construct the relationship inference model. Table I presents the average runtime and 95% confidence intervals for each model, demonstrating that the HDCG model is approximately 34 times more efficient than the DCG model under the assumptions of this experiment. The variation in average runtime is more significant for the HDCG model than for the DCG model because most of the computation of the DCG model is centered on grounding the meaning of the single verb phrases in each example.

TABLE I
MEAN INFERENCE TIME OF THE DCG AND HDCG MODELS WITH 95% CONFIDENCE INTERVALS.

Model	Mean Inference Time (sec)
DCG	0.485 (0.003)
HDCG	0.014 (0.002)

To better illustrate the behavior of the HDCG model, we examine the rules and groundings inferred from the sentence "approach the left of the table" in the context of an environment composed of four objects. This example did not exist in the training data and was evaluated in 0.023 seconds. The HDCG model inferred rules that permitted objects of type ROBOT, TABLE, and UNKNOWN, regions of type LEFT and UNKNOWN, and constraints of type INSIDE. These six rules reduced the number of symbols that needed to be considered for the noun and prepositional phrases and verb phrases in the relationship inference model to five and twenty-four, respectively. One symbol was inferred by the relationship inference model, which correctly consisted of a constraint of type INSIDE between the object of type ROBOT and the region of type LEFT with respect to the object with the type TABLE.

V. DISCUSSION

Improving the computational efficiency of models for symbol grounding is important for the widespread adoption of natural language interfaces for service robotics. The ability to understand complicated utterances that describe non-trivial tasks in complex environments is necessary to enable users of assistive technologies to regain their independence in diverse scenarios. We have demonstrated that the HDCG model can improve the computational efficiency with which we can infer the meaning of instructions by more than an order of magnitude over the DCG model. Learning the properties of search spaces in addition to the meaning of particular phrases enables more efficient search by eliminating constituents that do not influence the resulting distribution of symbols. The ratio of this improvement over the DCG model is proportional to the size of the search spaces and the sparsity of constituents in annotated examples. In future work, we will investigate the impact of more sophisticated hierarchies of graphical models and their impact on probabilistic inference. Since the symbols required to represent instructions (e.g., "get me the coffee mug from the cupboard") may differ significantly from those that are simply observations of the physical environment (e.g., "there is a coffee mug in the cupboard to the right of the refrigerator"), it may be beneficial to learn a hierarchy composed of more than two layers where rules for how to partition subsequent spaces of rules are part of the inference procedure. We are also exploring ways in which to infer (rather than assume) the conditional independence assumptions and relationships between constituents when constructing each probabilistic graphical model, and migrating from semantic tags towards learning objects from physical object properties (e.g. color, mass, shape).

Advancements to natural language interfaces have many applications in healthcare and assistive robotics beyond smart wheelchairs. Natural language interfaces could be integrated into wearable rehabilitation devices to allow non-expert operators to control a wider range of therapies that simulate common tasks. Efficient models for natural language understanding could also be inverted to provide realtime feedback to users of rehabilitation devices and assist in perception for the visually impaired. In future work, we plan to apply the HDCG model for natural language understanding of robot instructions and environment observations on a smart wheelchair and evaluate the performance in the context of assistive robotics tasks.

VI. ACKNOWLEDGMENT

This work was partially supported by the Robotics Consortium of the U.S Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement W911NF-10-2-0016, and by the Office of Naval Research under MURI N00014-09-1-1052.

REFERENCES

[1] F. Doshi-Velez, W. Li, Y. Battat, B. Charrow, D. Curthis, J. Park, S. Hemachandra, J. Velez, C. Walsh, D. Fredette, B. Reimer, N. Roy, and S. Teller, "Improving safety and operational efficiency in residential care settings with wifi-based localization," *Journal of the American Medical Directors Association*, vol. 13, no. 6, pp. 558–563, 2013.

[2] H. Krebs, B. Volpe, M. Aisen, and N. Hogan, "Increasing productivity and quality of care : Robot-aided neuro-rehabilitation," *Journal of Rehabilitation Research and Development*, vol. 37, no. 6, pp. 639–652, 2000.

[3] A. Roy, H. Krebs, D. Williams, C. Bever, L. Forrester, R. Macko, and N. Hogan, "Robot-aided neurorehabilitation: A novel robot for ankle rehabilitation," *IEEE Trans. On Robotics*, vol. 25, no. 3, pp. 569 – 582, 2009.

[4] R. Simpson, "Smart wheelchairs: a literature review," *Journal of Rehabilitation Research and Development*, vol. 42, no. 4, p. 423436, 2005.

[5] G. Bourhis and M. Sahnoun, "Assisted control mode for a smart wheelchair," in *Proc. of the 2007 IEEE 10th Int'l Conf. on Rehabilitation Robotics*, 2007.

[6] R. Barea, L. Boquete, M. Mazo, E. Lpez, and L. Bergasa, "E.o.g. guidance of a wheelchair using neural networks," in *Proc. 15th Int'l Conf. on Pattern Recognition*, 2000, pp. 668 – 671.

[7] T. Carlson and J. D. R. Millan, "Braincontrolled wheelchairs: A robotic architecture," *IEEE Robotics and Automation Magazine*, vol. 20, no. 1, pp. 65–73, March 2013.

[8] Y. Kuno, N. Shimada, and Y. Shirai, "A robotic wheelchair based on the integration of human and environmental observations," *IEEE Robotics and Automation Magazine*, vol. 10, no. 1, p. 2634, March 2003.

[9] M. MacMahon, B. Stankiewicz, and B. Kuipers, "Walk the talk: Connecting language, knowledge, and action in route instructions," in *Proc. Nat'l Conf. on Artificial Intelligence*, 2006.

[10] T. Kollar, S. Tellex, D. Roy, and N. Roy, "Toward understanding natural language directions," in *Proc. ACM/IEEE Int'l Conf. on Human-Robot Interaction*, Osaka, Japan, 2010, pp. 259–266.

[11] D. Chen and R. Mooney, "Learning to interpret natural language navigation instructions from observations," in *Proc. Nat'l Conf. on Artificial Intelligence*, August 2011, pp. 859–865.

[12] C. Matuszek, E. Herbst, L. Zettlemoyer, and D. Fox, "Learning to parse natural language commands to a robot control system," in *Proc. Int'l Symp. on Experimental Robotics*, 2012.

[13] J. Fasola and M. Mataric, "Using semantic fields to model dynamic spatial relations in a robot architecture for natural language instruction of service robots," in *Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, November 2013, pp. 143 – 150.

[14] M. R. Walter, S. Hemachandra, B. Homberg, S. Tellex, and S. Teller, "Learning semantic maps from natural language descriptions," in *Proc. Robotics: Science and Systems*, 2013.

[15] S. Tellex, T. Kollar, S. Dickerson, M. R. Walter, A. G. Banerjee, S. Teller, and N. Roy, "Understanding natural language commands for robotic navigation and mobile manipulation," in *Proc. Nat'l Conf. on Artificial Intelligence*, San Francisco, CA, 2011.

[16] C. Matuszek, L. Bo, L. Zettlemoyer, and D. Fox, "Learning from unscripted deictic gesture and language for human-robot interactions," in *Proc. AAAI Conf. on Artificial Intelligence*, 2014.

[17] R. Mooney, "Learning to connect language and perception," in *Proc. AAAI Conf. on Artificial Intelligence*, July 2008, pp. 1598–1601.

[18] T. Howard, S. Tellex, and N. Roy, "A natural language planner interface for mobile manipulators," in *Proc. Int'l Conf. on Robotics and Automation*, 2014.

[19] F. Duvallet, M. Walter, T. Howard, S. Hemachandra, J. Oh, S. Teller, N. Roy, and A. Stentz, "Inferring maps and behaviors from natural