Driving Under the Influence (of Language)

Daniel Paul Barrett and **Scott Alan Bronikowski** and **Haonan Yu** and **Jeffrey Mark Siskind**
Purdue University
School of Electrical and Computer Engineering
465 Northwestern Avenue
West Lafayette, IN 47907-2035, USA
{dpbarret,sbroniko,yu239,qobi}@purdue.edu

Abstract

We present a unified framework which supports grounding natural-language semantics in robotic driving. This framework supports acquisition (learning grounded meanings of nouns and prepositions from human annotation of robotic driving paths), generation (using such acquired meanings to generate sentential description of new robotic driving paths), and comprehension (using such acquired meanings to support automated driving to accomplish navigational goals specified in natural language). We evaluate the performance of these three tasks by having human judges rate the semantic fidelity of the sentences associated with paths, achieving overall average correctness of 94.6% and overall average completeness of 85.6%.

1 Introduction

With recent advances in machine perception and robotic automation, it becomes increasingly relevant and important to allow machines to interact with humans in natural language in a *grounded fashion*, where the language refers to actual things and activities in the world. Here, we present our efforts to automatically drive—and learn to drive—a mobile robot under natural-language command. Our contribution is summarized in Fig. 1. A human teleoperator drives a mobile robot under radio control through a variety of floorplans. A wireless video feed allows such to be done without direct line-of-sight view of the terrain by the teleoperator. Onboard sensors and computation can determine the location of the robot by odometry and inertial guidance in real time. This allows the robot to log traces of the driving path. Humans can then annotate such with sentential descriptions. From a training corpus of paths paired with sentential descriptions and floorplan specifications, our system can automatically learn the meanings of nouns that refer to objects in the floorplan and prepositions that describe spatial relations between such objects, as well as the path taken by the robot. With such learned meanings, the robot can then generate sentential descriptions of new driving activity undertaken by the teleoperator. Moreover, instead of manually controlling the robot through teleoperation, one can issue the robot natural-language commands which can induce fully automatic driving without human assistance to satisfy the path specified in the natural-language command.

We have conducted experiments with an actual radio-controlled robot that demonstrate all three of these modes of operation: acquisition, generation, and comprehension. We demonstrate successful completion of all three of these tasks on hundreds of driving examples. We evaluate the fidelity of the sentential descriptions produced automatically in response to manual driving and the fidelity of the driving paths induced automatically to fulfill natural-language commands, by presenting the pairs of sentences together with the associated paths to human judges. Overall, the average "correctness" (the degree to which the description is true of the path) reported is 94.6% and the average "completeness" (the degree to which the description fully covers the path) reported is 85.6%.

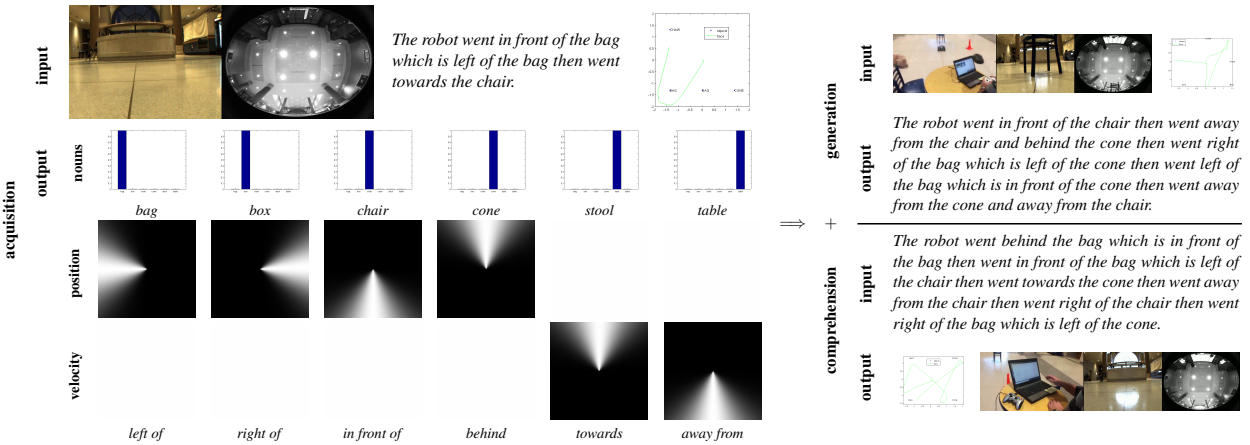

Figure 1: (left) A human teleoperator drives the mobile robot through 250 paths with a live video feed from front-facing and omnidirectional cameras. Odometry reconstructs the paths taken by the robot. A human annotates each path with an English description. This allows the robot to learn the meanings of the nouns and prepositions. Hand-designed models are shown here for reference; actual learned models are shown in Fig. 7. Note that the distributions are uniform in velocity angle (bottom row) for *left of*, *right of*, *in front of*, and *behind* and in position angle (top row) for *towards* and *away from*. These learned meanings support generation of English descriptions of new paths driven by teleoperation (top right) and autonomous unassisted driving of paths that meet navigational goal specified in English descriptions (bottom right).

2 Technical Details

2.1 Grammar and Logical Form

We employ a small fixed handwritten grammar as shown in Fig. 2. Nothing turns on this however. In principle, one could replace this grammar with any other mechanism for generating logical form (Bos et al., 2004; Clark and Curran, 2003, 2007; Ge and Mooney, 2006; Goldwasser et al., 2011; Kate et al., 2005; Kate and Mooney, 2006; Kwiatkowski et al., 2010, 2011; Liang et al., 2009, 2013; Lu et al., 2008; Miller et al., 1996; Nguyen et al., 2006; Och and Ney, 2003; Papineni et al., 1997; Poon and Domingos, 2009, 2010; Ramaswamy and Kleindienst, 2000; Tang and Mooney, 2000; Thompson and Mooney, 2003; Vogel and Jurafsky, 2010; Watkinson and Manandhar, 2001; Wong and Mooney, 2006, 2007; Zelle and Mooney, 1996; Zettlemoyer and Collins, 2005, 2007, 2009). This paper concerns itself with semantics, not syntax, and only addresses issues relating to the grounding of logical form. This particular grammar is simply a convenient surface representation of our logical form.

Note that our surface syntax allows two uses of prepositions (and the associated prepositional phrases): as modifiers to nouns in noun phrases, indicated with a subscript 'SR' (*i.e.,* spatial relation), and as adjuncts to verbs in verb phrases, indicated with a subscript 'path.' Many prepositions can be used in both SR and path form. They share the same semantic representation and both uses are learned from the pooled data of both kinds of occurrences in the training corpus. Furthermore, note that the grammar supports infinite NP recursion: noun phrases can contain prepositional phrases that, in turn, contain noun phrases. Finally, note that the grammar supports conjunctions of prepositional phrases in both SR and path form.

$$
\begin{array}{lcl}
\mathrm{S} & \rightarrow & \textit{The robot}\ \mathrm{VP} \\
\mathrm{VP} & \rightarrow & \textit{went}\ \mathrm{PP}_{\mathrm{path}}\ [\textit{then}\ \mathrm{VP}] \\
\mathrm{PP}_{\mathrm{path}} & \rightarrow & \mathrm{P}_{\mathrm{path}}\ \mathrm{NP}\ [\textit{and}\ \mathrm{PP}_{\mathrm{path}}] \\
\mathrm{NP} & \rightarrow & \textit{the}\ \mathrm{N}\ [\mathrm{PP}_{\mathrm{SR}}] \\
\mathrm{PP}_{\mathrm{SR}} & \rightarrow & \textit{which is}\ \mathrm{P}_{\mathrm{SR}}\ \mathrm{NP}\ [\textit{and}\ \mathrm{PP}_{\mathrm{SR}}] \\
\mathrm{P}_{\mathrm{path}} & \rightarrow & \textit{left of} \mid \textit{right of} \mid \textit{in front of} \mid \textit{behind} \\
 & \mid & \textit{towards} \mid \textit{away from} \\
\mathrm{P}_{\mathrm{SR}} & \rightarrow & \textit{left of} \mid \textit{right of} \mid \textit{in front of} \mid \textit{behind} \\
\mathrm{N} & \rightarrow & \textit{bag} \mid \textit{box} \mid \textit{chair} \mid \textit{cone} \mid \textit{stool} \mid \textit{table}
\end{array}
$$

Figure 2: The grammar used by our implementation.

We employ the logical form shown in Fig. 3. Informally, formulas in logical form denote paths through a floorplan. Both paths and floorplans are

$$
\begin{array}{lcl}
\langle \textit{formula} \rangle & \rightarrow & \langle \textit{path quantifier} \rangle \langle \textit{floorplan quantifier} \rangle \\
 & & \langle \textit{atomic formula} \rangle (\wedge \langle \textit{atomic formula} \rangle)^* \\
\langle \textit{path quantifier} \rangle & \rightarrow & [\langle \textit{var} \rangle (; \langle \textit{var} \rangle)^*] \\
\langle \textit{floorplan quantifier} \rangle & \rightarrow & \{\langle \textit{var} \rangle (, \langle \textit{var} \rangle)^*\} \\
\langle \textit{atomic formula} \rangle & \rightarrow & \langle \textit{atomic formula}_1 \rangle \\
 & | & \langle \textit{atomic formula}_2 \rangle \\
\langle \textit{atomic formula}_1 \rangle & \rightarrow & \textsc{bag}(\langle \textit{var} \rangle) \\
 & | & \textsc{box}(\langle \textit{var} \rangle) \\
 & | & \textsc{chair}(\langle \textit{var} \rangle) \\
 & | & \textsc{cone}(\langle \textit{var} \rangle) \\
 & | & \textsc{stool}(\langle \textit{var} \rangle) \\
 & | & \textsc{table}(\langle \textit{var} \rangle) \\
\langle \textit{atomic formula}_2 \rangle & \rightarrow & \textsc{leftOf}(\langle \textit{var} \rangle, \langle \textit{var} \rangle) \\
 & | & \textsc{rightOf}(\langle \textit{var} \rangle, \langle \textit{var} \rangle) \\
 & | & \textsc{inFrontOf}(\langle \textit{var} \rangle, \langle \textit{var} \rangle) \\
 & | & \textsc{behind}(\langle \textit{var} \rangle, \langle \textit{var} \rangle) \\
 & | & \textsc{towards}(\langle \textit{var} \rangle, \langle \textit{var} \rangle) \\
 & | & \textsc{awayFrom}(\langle \textit{var} \rangle, \langle \textit{var} \rangle)
\end{array}
$$

Figure 3: The logical form used by our implementation.

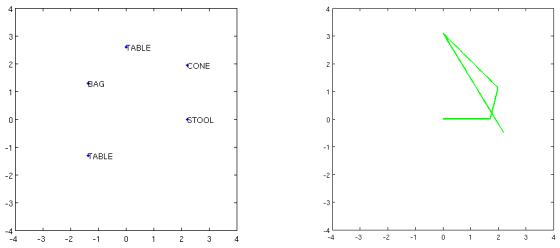

Figure 4: (left) A sample floorplan consisting of a set of labeled waypoints. (right) A sample path rendered as a curve of interpolated points passing through a sequence of unlabeled waypoints.

specified as collections of waypoints. A *waypoint* is a 2D Cartesian coordinate optionally labeled with the class of the object that resides at that coordinate, *e.g.,* $(3, 47, \textbf{bag})$ The waypoint is unlabeled, *e.g.,*$(3, 47)$, if no object resides at that coordinate. A *floorplan* is a set of labeled waypoints (Fig.4 left). A *path* is a sequence of unlabeled waypoints (Fig.4 right). A formula in logical form contains three parts: a *path quantifier*, a *floorplan quantifier*, and a *condition* that the path through the floorplan must satisfy. The condition is a conjunction of atomic formulas, predicates applied to variables bound by the path or floorplan quantifiers. The formula must be closed, *i.e.,* every variable in the condition must appear either in the path quantifier or the floorplan quantifier. The model of a formula is a set of bindings for each of the quantified path variables to unlabeled waypoints, and floorplan variables to labeled waypoints.

The one-argument atomic formulas constrain the class of waypoints to which the variables that appear as their arguments are bound. The two-argument atomic formulas constrain the spatial relations between pairs of waypoints to which the variables that appear as their arguments are bound. The logical form in Fig. 3 contains a particular set of six one-argument predicate and six two-argument predicates. Nothing turns on this however. This is simply the set of predicates that we use in the experiments reported. The framework clearly extends to any number of predicates of any arity, particularly since we learn the meanings of the predicates.

Straightforward (semantic) parsing and surface generation techniques map bidirectionally between the surface language form as specified by the grammar in Fig. 2 and the logical form in Fig. 3. For example, a surface form like

> *The robot went towards the stool, then went behind the chair which is right of the stool, then went towards the cone, then went away from the chair which is left of the cone, then went in front of the table.*

(commas added for legibility) would correspond to the following logical form:

$$
[\alpha, \beta, \gamma, \delta, \epsilon]\{t, u, v, w, x, y, z\}
\left(
\begin{array}{l}
\textsc{towards}(\alpha, t) \wedge \textsc{stool}(t) \wedge \\
\textsc{behind}(\beta, u) \wedge \textsc{chair}(u) \wedge \textsc{rightOf}(u, v) \wedge \textsc{stool}(v) \wedge \\
\textsc{towards}(\gamma, w) \wedge \textsc{cone}(w) \wedge \\
\textsc{awayFrom}(\delta, x) \wedge \textsc{chair}(x) \wedge \textsc{leftOf}(x, y) \wedge \textsc{cone}(y) \wedge \\
\textsc{inFrontOf}(\epsilon, z) \wedge \textsc{table}(z)
\end{array}
\right)
$$

Note that in the above, nouns all correspond to one-argument predicates while one-argument prepositions all correspond to two-argument predicates. But nothing turns on this. One could imagine lexical prepositional phrases, like *leftward*, that correspond to one-argument predicates. Moreover, path uses of prepositions specify waypoints in the path. These appear in logical form as predicates whose first argument is a variable in the path quantifier. Similarly, SR uses of prepositions specify waypoints in the floorplan. These appear in logical form as predicates whose first argument is a variable in the floorplan quantifier. Thus, in the above, the atomic formulas $\textsc{towards}(\alpha, t)$, $\textsc{behind}(\beta, u)$, $\textsc{towards}(\gamma, w)$, $\textsc{awayFrom}(\delta, x)$, and $\textsc{inFrontOf}(\epsilon, z)$ constitute path uses while the atomic formulas $\textsc{rightOf}(u, v)$ and $\textsc{leftOf}(x, y)$ constitute SR uses. Also note that each (path) prepositional phrase consists of a subset

of the atomic formulas in the condition, as indicated above by the line breaks.

2.2 Representation of the Lexicon

The lexicon specifies the meanings of the one- and two-argument predicates in logical form. The meanings of one-argument predicates are discrete distributions over the set of class labels. Note that the one-argument predicates, like BAG, are distinct from the class labels, like **bag**. The mapping between such is learned. Moreover, a given floorplan might have multiple instances of objects of the same class. These would be disambiguated with complex noun phrases such as *the chair which is right of the stool* and *the chair which is left of the cone*. Such disambiguating prepositional phrase modifiers of noun phrases can be nested and conjoined arbitrarily. Similarly, waypoints can be disambiguated by conjunctions of prepositional phrase adjuncts.

Two-argument predicates specify relations between target objects and reference objects. In SR uses, the reference object is the object of the preposition while the target object is the head noun. For example, in *the chair to the left of the table*, *chair* is the target object and *table* is the reference object. In path uses, the target object is a waypoint in the robot path while the reference object is the object of the preposition. For example, in *went towards the table*, *table* is the reference object. The lexical entry for each two-argument predicate is specified as the location μ and concentration κ parameters for multiple independent von Mises distributions for a variety of angles between target and reference objects.

The meanings of two-argument predicates are specified as a pair of von Mises distributions on angles. One, the *position angle*, is the orientation of a vector from the coordinates of the reference object to the coordinates of the target object (Fig. 5 left).[1] The same distribution is used both for SR and path uses. The second, the *velocity angle*, is the angle between the velocity vector at a waypoint and a vector from the coordinates of the waypoint to the coordinates of the reference object (Fig. 5 right). This is only used for path uses, because it requires computation of the direction of robot motion which is determined from adjacent waypoints in the path. This angle is thus taken from the frame of reference of the robot.

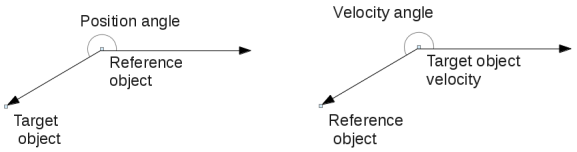

Figure 5: (left) How position angles are measured. (right) How velocity angles are measured.

Fig. 1(bottom left) illustrates how this framework is used to represent the meanings of one-argument prepositions. Here, we render the angular distributions as potential fields around the reference object at the center for the position angle, and the target object at the center for the velocity angle. The intensity of a point (target object for position angle) reflects its probability mass. Note that the distributions are uniform in velocity angle for *left of*, *right of*, *in front of*, and *behind* and in position angle for *towards* and *away from*.

2.3 Tasks

We formulate sentential semantics as a variety of relationships between a sentence $\mathbf{s}$, or more precisely a formula in logical form, a path $\mathbf{p}$, a sequence of unlabeled waypoints, a floorplan $\mathbf{f}$, a set of labeled waypoints, and a lexicon Λ, the collective μ and κ parameters for the angular distributions for each of the two-argument predicates and the discrete distributions for each of the one-argument predicates.

generation Generate a sentence $\mathbf{s}$ that describes an observed path $\mathbf{p}$ taken by the robot in a given floorplan $\mathbf{f}$ with a known lexicon Λ.

comprehension Generate a path $\mathbf{p}$ to be taken by the robot that satisfies a given sentence $\mathbf{s}$ issued as a command in a given floorplan $\mathbf{f}$ with a known lexicon Λ.

acquisition Learn a lexicon Λ from a collection of observed paths $\mathbf{p}_i$ taken by the robot in the corresponding floorplans $\mathbf{f}_i$ as described by the corresponding sentences $\mathbf{s}_i$ provided by human annotators.

[1] Without loss of generality, this assumes an implicit frame of reference taken to be that of an exogenous person observing the path located at the bottom of the floorplan. It would be straightforward to rotate this angle to put it in the perspective of any known observer position.

2.3.1 Generation

When generating language, the path $\mathbf{p}$ is obtained by odometry that measures the path taken by the robot when driven by a human operator under remote wireless control. This path consists of a collection of 2D floor positions densely sampled at 50Hz. To generate a formula in logical form, and thus the corresponding sentence, one must select a subsequence of this dense sequence worthy of description.

During generation, we care about three properties: "correctness," that the sentence be logically true of the path, "completeness," that the sentence differentiate the intended path from all other possible paths, and "conciseness," that the sentence be the shortest that does so. We attempt to find a balance between these properties with the following heuristic algorithm. First we downsample the path by computing the integral distance traveled from the initial position for each point in the dense path and selecting a subsequence whose points are separated by 5cm of integral path length. We then produce a path prepositional phase to describe each path waypoint by selecting that atomic formula with maximum posterior probability constructed out of a two-argument predicate with the path waypoint as its first argument and with a floorplan waypoint as its second argument. Identical such choices for consecutive sets of waypoints in the path are coalesced. We then generate a noun phrase for the object of each waypoint preposition that refers to that referenced floorplan waypoint. This is done by searching the space of noun phrases generated by the grammar in Fig. 2 in order of increasing length (to satisfy conciseness) that are true of that floorplan waypoint (to satisfy correctness) and that are false of all other floorplan waypoints (to satisfy completeness). When doing so, we take a one-argument predicate to be true of that class with maximum posterior probability and false of all others. Similarly, for each pair of floorplan waypoints, we take that two-argument predicate with maximum posterior probability to be true of that tuple and all other predicates applied to that tuple to be false. Thus when the floorplan contains a single instance of a class, it can be referred to with a simple noun. But when there are multiple instances of a class, the shortest possible noun phrase, with one or more SR prepositional phrases, is generated to disambiguate.

2.3.2 Comprehension

To perform comprehension, we formulate a scoring function $\mathcal{R}(\mathbf{s}, \mathbf{p}, \mathbf{f}, \Lambda)$ over an unknown path $\mathbf{p}$ and optimize this scoring function

$$\mathbf{p}^* = \arg \max_{\mathbf{p}} \mathcal{R}(\mathbf{s}, \mathbf{p}, \mathbf{f}, \Lambda)$$

using gradient ascent. This scoring function takes a formula in logical form derived from $\mathbf{s}$ and internally maximizes over all possible bindings from variables in the floorplan quantifier to waypoints in the floorplan $\mathbf{f}$. The objective of this internal maximization is the product of the probability determined by each atomic formula in the formula derived from $\mathbf{s}$, given the probability models for the predicates as specified by the parameters in Λ, essentially computing a MAP estimate of the joint probability of satisfying the conjunction of atomic formulas assuming that they are independent.

The above scoring function alone is insufficient. It can produce path waypoints that are too close together. To remedy this, a penalty term is added for each pair of path waypoints to drive them away from each other. It also can produce paths that get too close to floorplan waypoints. To remedy this, the same penalty term is added between each pair of a path waypoint and a floorplan waypoint to drive them away from each other. Finally, our formulation of the semantics of prepositions is based on angles but not distance. Thus there is is a large subspace of the floor that leads to equal probability of satisfying each atomic formula, *i.e.,* the cones in Fig. 1. This allows a path to satisfy a prepositional phrase like *to the left of the chair* by being far away from the chair. To remedy this, we add a small penalty term between each path waypoint and the floorplan waypoints selected as its reference objects to prefer short distances. This third penalty term is smaller than the others because it serves just to satisfy completeness while the others serve to prevent collisions or prevent local optima in the cost function. A postprocessing step performs obstacle avoidance by adding additional path waypoints as needed.

2.3.3 Acquisition

To perform acquisition, we formulate a large hidden Markov model (HMM), with a state for every

path prepositional phrase in each sentence in the training corpus. The observations for this HMM are the sequences of path waypoints in the training corpus. The output model for a given state is the same scoring function as used for comprehension except that it is limited to only that path prepositional phrase and computes a likelihood over all mappings from variables in the floorplan quantifier to floorplan waypoints instead of a MAP estimate. The transition function for the HMM allows each state to self loop or transition to the state for the next path prepositional phrase in the training sentence. No other transitions are allowed. The HMM is constrained to start in the state associated with the first path prepositional phrase in the sentence associated with each path. We add dummy states, with a small fixed output probability, between the states for each pair of adjacent path prepositional phrases, as well as at the beginning and end of each sentence, to allow for portions of the path that are not described in the associated sentence. We then train this HMM with Baum-Welch (Baum and Petrie, 1966; Baum et al., 1970; Baum, 1972) and discard the learned transition matrix. This trains the distributions for the words in the lexicon Λ as they are tied as components of the output models.

3 Our Mobile Robot

The experiments reported here were performed on a custom mobile robot (Fig. 6). This robot can be driven by a human teleoperator and automatically driven to accomplish specified navigational goals. When conducting experiments on generation and acquisition, a human teleoperator drives the robot along a variety of paths in a variety of floorplans. During such teleoperation, the driver controls the robot with a game controller and receives real-time video feedback from both a forward-facing camera mounted on a pan-tilt unit and an upward facing omnidirectional camera. Bidirectional communication during such teleoperation is performed over WiFi or 4G LTE. During all operation, robot localization is performed onboard the robot in real-time with odometry information from shaft encoders on the wheels and inertial-guidance information from an IMU. The video feed, localization, and all sensor and actuator data is logged in a time-stamped format. The path recovered from localization can support generation and acquisition. When conducting experiments on comprehension, the path is planned automatically; the robot can then automatically follow such a path by comparing the new odometry gathered in real time with the planned path and adjusting the wheel rotational velocities accordingly.

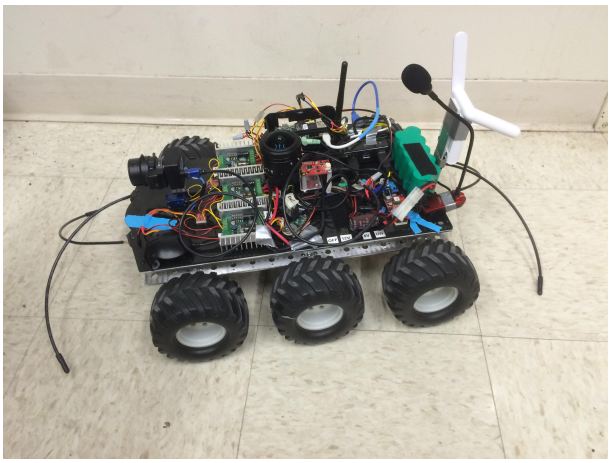

Figure 6: Our custom mobile robot.

4 Experiments

We conducted an experiment as outlined in Fig. 1. We generated 250 random sentences from the grammar in Fig. 2, 25 in each of 10 different floorplans that were randomly generated to place either 4 or 5 objects, with 2 objects always being of the same class, to introduce ambiguity requiring disambiguation via SR prepositional phrases, at one of 12 possible grid positions. Path data was logged while a human teleoperator manually drove the robot to comply with these sentential instructions in these floorplans (Fig. 7 top). Models were learned for each of the nouns and prepositions. These were used to automatically generate descriptions for 10 different new paths manually driven by a human teleoperator in 10 new random floorplans (Fig. 7 middle). These were also used to automatically drive the robot unassisted to follow 10 different new random sentences in each of 10 different new random floorplans where the same objects could be placed at one of 56 possible grid positions (Fig. 7 bottom). The random sentences used for training had either 2 or 3 path waypoints while those used for generation and comprehension had either 5 or 6 path waypoints.

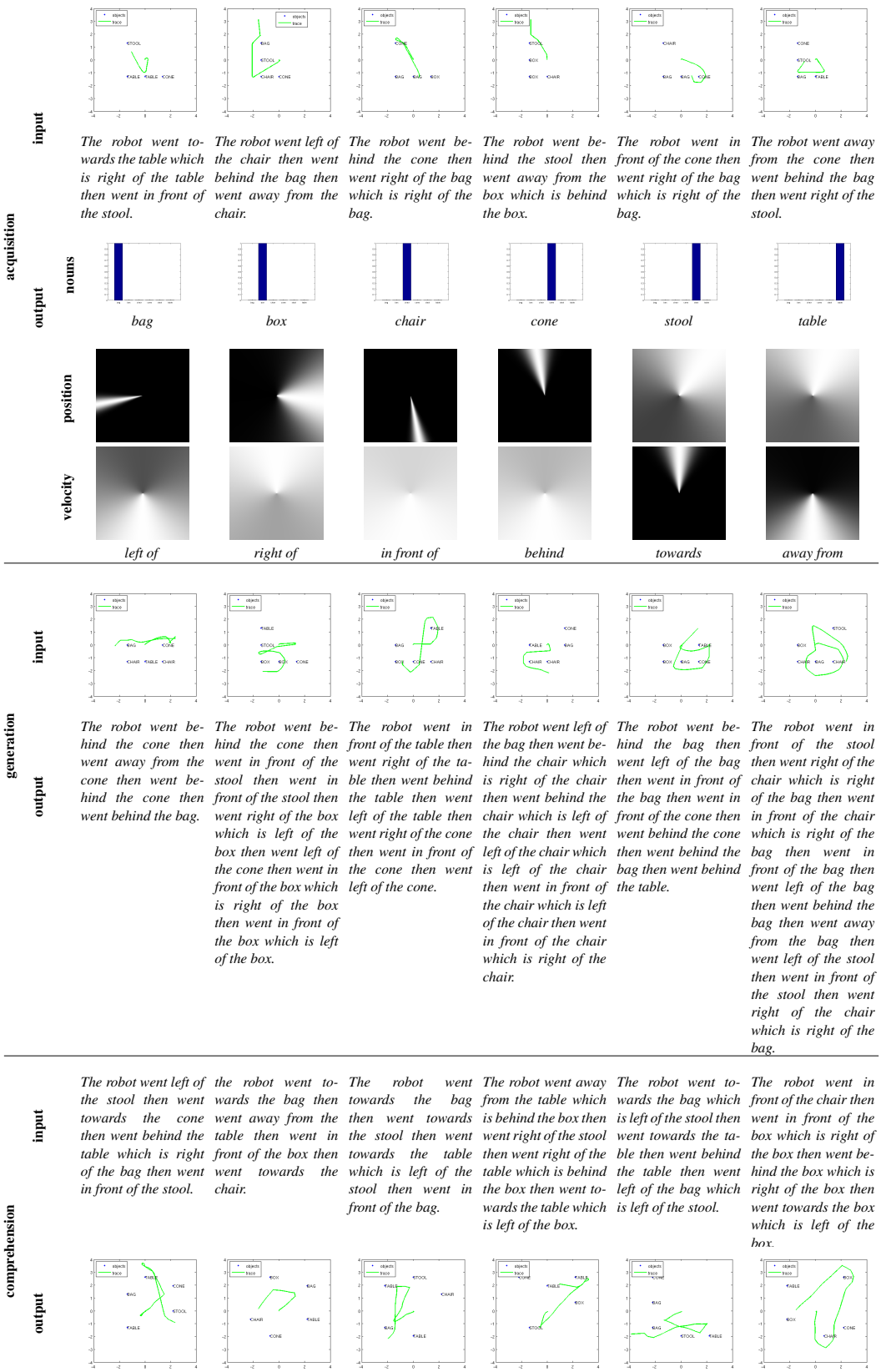

Figure 7: Example experimental runs, 6 out of 250 for acquisition and 100 for each of generation and comprehension. Videos available at `http://naacl-hlt2015-sub.blogspot.com/`.

Odometry and inertial guidance were used to determine paths driven. Pairs of sentences and paths obtained during both generation and comprehension were given to a pool of 6 independent judges to obtain 3 judgments on each. Judges were asked to label each path prepositional phrase in each sentence paired with the entire path as being either 'correct' or 'incorrect', *i.e.,* whether it was true of the intended portion of the path as determined by that judge. For generation, judges were also asked to assess how much of the path was described by the sentence, giving a completeness judgment ranging from 0 (worst) to 5 (best). These were converted to percentages. For comprehension, judges were also asked to assess what fraction of the path constitutes motion that is described by the sentence (quantized as 0 to 5). These were again converted to percentages to measure completeness. For generation, judgments were obtained twice, pairing each input path with sentences generated using the hand-constructed models from Fig. 1 as well the learned models from Fig. 7. For comprehension, judgments were also obtained twice, pairing each input sentence with both the planned path as well as the actually driven path as determined by odometry and inertial guidance. Fig. 8(top) summarizes the judgments aggregated across the 3 judges and 100 samples. The standard deviations are across the mean value of the 3 judges for each sample. Overall, the average "correctness" reported is 94.6% and the average "completeness" reported is 85.6%.

For generation, we also measured "conciseness" by having the 3 human judges score each generated sentence as -2 (much too short), -1 (too short), 0 (about right), 1 (too long), or 2 (much too long). Fig. 8(bottom) summarize these judgments as histograms. Overall, judges assessed that the generated sentence length was 'about right' a little over half of the time, with generation erring more towards being too long than too short.

	correctness		completeness	
	mean	std dev	mean	std dev
generation (hand-constructed models)	94.6%	4.54%	85.5%	2.26%
generation (learned models)	92.0%	6.11%	84.2%	6.35%
comprehension (planned path)	96.2%	0.38%	88.5%	11.5%
comprehension (measured path)	95.5%	1.42%	84.7%	9.9%

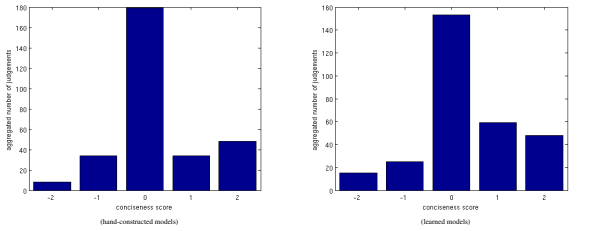

Figure 8: Correctness, completeness, and conciseness results of human evaluation of sentences automatically generated from manually driven paths and automatically driven paths produced by comprehension of provided sentences.

5 Related Work

This paper builds on the work of many previous authors, agglomerating several disparate ideas to synthesize a new direction. Like Chen and Mooney (2011), we use examples of motion paired with sentential descriptions of that motion to learn models for the words in the descriptions, although our models are more closely related to those in Yu and Siskind (2013). We use untrained outside observers to evaluate the correctness of our output, as in Tellex et al. (2011). The set of commands that our system is able to understand is necessarily limited to a subset of the English language, like Teller et al. (2010). Our breakdown of the instruction sentence into spatially-relevant phrases is similar to the spatial description clauses presented in Kollar et al. (2010). As with the MARCO system presented in MacMahon et al. (2006), our system possesses the ability to follow natural-language instructions of infinite variety, constrained only by the grammar in Fig. 2. Our robot performs localization through the use of wheel odometry in conjunction with a gyro, as in Azizi and Houshangi (2004), although we use an Extended Kalman Filter (Jazwinski, 1970) instead of the Unscented Kalman Filter (Wan and Van Der Merwe, 2000).

6 Conclusion

We have demonstrated a novel approach for grounding the semantics of natural language in the domain of mobile-robot navigation. Sentences describe paths taken by the robot relative to other objects in the environment. The meanings of nouns and prepositions are trained from a corpus of paths driven by a human teleoperator annotated with sentential descriptions. These can then support both au-

tomatic generation of sentential descriptions of new paths driven as well as automatic driving of paths to satisfy navigational goals specified in provided sentences. This is a step towards the ultimate goal of grounded natural language that allows machines to interact with humans when the language refers to actual things and activities in the real world.

Acknowledgments

This research was sponsored, in part, by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-10-2-0060. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either express or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes, notwithstanding any copyright notation herein.

References

Azizi, F. and Houshangi, N. (2004). Mobile robot position determination using data from gyro and odometry. In *Canadian Conference on Electrical and Computer Engineering*, volume 2, pages 719–722.

Baum, L. E. (1972). An inequality and associated maximization technique in statistical estimation of probabilistic functions of a Markov process. *Inequalities*, 3:1–8.

Baum, L. E. and Petrie, T. (1966). Statistical inference for probabilistic functions of finite state Markov chains. *The Annals of Mathematical Statistics*, 37:1554–63.

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41(1):164–71.

Bos, J., Clark, S., Steedman, M., Curran, J. R., and Hockenmaier, J. (2004). Wide-coverage semantic representations from a CCG parser. In *International Conference on Computational Linguistics*, pages 1240–1247.

Branavan, S. R. K., Zettlemoyer, L. S., and Barzilay, R. (2010). Reading between the lines: Learning to map high-level instructions to commands. In *ACL*, pages 1268–1277.

Carpenter, B. (1997). *Type-logical semantics*. MIT Press.

Chen, D. L. and Mooney, R. J. (2011). Learning to interpret natural language navigation instructions from observations. In *AAAI*, pages 859–865.

Clark, S. and Curran, J. R. (2003). Log-linear models for wide-coverage CCG parsing. In *EMNLP*, pages 97–104.

Clark, S. and Curran, J. R. (2007). Wide-coverage efficient statistical parsing with CCG and log-linear models. *Computational Linguistics*, 33(4):493–552.

Clarke, J., Goldwasser, D., Chang, M.-W., and Roth, D. (2010). Driving semantic parsing from the world's response. In *Conference on Computational Natural Language Learning*, pages 18–27.

Ge, R. and Mooney, R. J. (2006). Discriminative reranking for semantic parsing. In *COLING-ACL*, pages 263–270.

Goldwasser, D., Reichart, R., Clarke, J., and Roth, D. (2011). Confidence driven unsupervised semantic parsing. In *ACL: Human Language Technologies*, pages 1486–1495.

He, Y. and Young, S. (2005). Semantic processing using the hidden vector state model. *Computer Speech & Language*, 19(1):85–106.

Hockenmaier, J. and Steedman, M. (2002). Generative models for statistical parsing with combinatory categorial grammar. In *ACL*, pages 335–342.

Jazwinski, A. H. (1970). *Stochastic Processes and Filtering Theory*. Academic Press.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering*, 82(1):35–45.

Kate, R. J. and Mooney, R. J. (2006). Using string-kernels for learning semantic parsers. In *COLING-ACL*, pages 913–920.

Kate, R. J., Wong, Y. W., and Mooney, R. J. (2005). Learning to transform natural to formal languages. In *AAAI*, pages 1062–1068.

Kollar, T., Tellex, S., Roy, D., and Roy, N. (2010). Toward understanding natural language directions. In *International Conference on Human-Robot Interaction*, pages 259–266.

Kwiatkowski, T., Zettlemoyer, L., Goldwater, S., and Steedman, M. (2010). Inducing probabilistic CCG grammars from logical form with higher-order unification. In *EMNLP*, pages 1223–1233.

Kwiatkowski, T., Zettlemoyer, L., Goldwater, S., and Steedman, M. (2011). Lexical generalization in CCG grammar induction for semantic parsing. In *EMNLP*, pages 1512–1523.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Liang, P., Jordan, M. I., and Klein, D. (2009). Learning semantic correspondences with less supervision. In *ACL-IJCNLP*, pages 91–99.

Liang, P., Jordan, M. I., and Klein, D. (2013). Learning dependency-based compositional semantics. In *ACL*, pages 389–446.

Lu, W., Ng, H. T., Lee, W. S., and Zettlemoyer, L. S. (2008). A generative model for parsing natural language to meaning representations. In *EMNLP*, pages 783–792.

MacMahon, M., Stankiewicz, B., and Kuipers, B. (2006). Walk the talk: connecting language, knowledge, and action in route instructions. In *AAAI*, pages 1475–1482.

Miller, S., Stallard, D., Bobrow, R., and Schwartz, R. (1996). A fully statistical approach to natural language interfaces. In *ACL*, pages 55–61.

Nguyen, L.-M., Shimazu, A., and Phan, X.-H. (2006). Semantic parsing with structured SVM ensemble classification models. In *COLING-ACL*, pages 619–626.

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Papineni, K. A., Roukos, S., and Ward, T. R. (1997). Feature-based language understanding. In *European Conference on Speech Communication and Technology*.

Poon, H. and Domingos, P. (2009). Unsupervised semantic parsing. In *EMNLP*, pages 1–10.

Poon, H. and Domingos, P. (2010). Unsupervised ontology induction from text. In *ACL*, pages 296–305.

Ramaswamy, G. N. and Kleindienst, J. (2000). Hierarchical feature-based translation for scalable natural language understanding. In *International Conference on Spoken Language Processing*, pages 506–509.

Siddharth, N., Barbu, A., and Siskind, J. M. (2014). Seeing what you're told: Sentence-guided activity recognition in video. In *CVPR*, pages 732–739.

Steedman, M. (1996). *Surface structure and interpretation*. MIT Press.

Steedman, M. (2000). *The syntactic process*. MIT Press.

Tang, L. R. and Mooney, R. J. (2000). Automated construction of database interfaces: Integrating statistical and relational learning for semantic parsing. In *Empirical Methods in Natural Language Processing and Very Large Corpora/ACL*, pages 133–141.

Teller, S., Walter, M. R., Antone, M., Correa, A., Davis, R., Fletcher, L., Frazzoli, E., Glass, J., How, J. P., Huang, A. S., et al. (2010). A voice-commandable robotic forklift working alongside humans in minimally-prepared outdoor environments. In *ICRA*, pages 526–533.

Tellex, S., Kollar, T., Dickerson, S., Walter, M. R., Banerjee, A. G., Teller, S. J., and Roy, N. (2011). Understanding natural language commands for robotic navigation and mobile manipulation. In *AAAI*, pages 1507–1514.

Thompson, C. A. and Mooney, R. J. (2003). Acquiring word-meaning mappings for natural language interfaces. *Journal of Artificial Intelligence Research*, 18(1):1–44.

Vogel, A. and Jurafsky, D. (2010). Learning to follow navigational directions. In *ACL*, pages 806–814.

Wan, E. A. and Van Der Merwe, R. (2000). The unscented Kalman filter for nonlinear estimation. In *Symposium on Adaptive Systems for Signal*

Processing, Communications, and Control, pages 153–158.

Watkinson, S. and Manandhar, S. (2001). Unsupervised lexical learning with categorial grammars using the LLL corpus. In *Learning Language in Logic*, pages 218–233.

Wong, Y. W. and Mooney, R. J. (2006). Learning for semantic parsing with statistical machine translation. In *NAACL-HLT*, pages 439–446.

Wong, Y. W. and Mooney, R. J. (2007). Learning synchronous grammars for semantic parsing with lambda calculus. In *ACL*, pages 960–967.

Yu, H. and Siskind, J. M. (2013). Grounded language learning from video described with sentences. In *ACL*, pages 53–63.

Zelle, J. M. and Mooney, R. J. (1996). Learning to parse database queries using inductive logic programming. In *AAAI*, pages 1050–1055.

Zettlemoyer, L. S. and Collins, M. (2005). Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *UAI*, pages 658–666.

Zettlemoyer, L. S. and Collins, M. (2007). Online learning of relaxed CCG grammars for parsing to logical form. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 678–687.

Zettlemoyer, L. S. and Collins, M. (2009). Learning context-dependent mappings from sentences to logical form. In *ACL-IJCNLP*, pages 976–984.