Toward Understanding Natural Language Directions

Thomas Kollar*
The Stata Center, MIT CSAIL
32 Vassar St, 32-331
Cambridge, MA 02139
tkollar@mit.edu

Stefanie Tellex*
MIT Media Lab
75 Amherst St. E14-574M
Cambridge, MA, 02139
stefie10@media.mit.edu

Deb Roy
MIT Media Lab
75 Amherst St, E14-574G
Cambridge, MA 02139
dkroy@media.mit.edu

Nicholas Roy
The Stata Center, MIT CSAIL
32 Vassar St, 32-330
Cambridge, MA 02139
nickroy@mit.edu

***Abstract*—Speaking using unconstrained natural language is an intuitive and flexible way for humans to interact with robots. Understanding this kind of linguistic input is challenging because diverse words and phrases must be mapped into structures that the robot can understand, and elements in those structures must be grounded in an uncertain environment. We present a system that follows natural language directions by extracting a sequence of *spatial description clauses* from the linguistic input and then infers the most probable path through the environment given only information about the environmental geometry and detected visible objects. We use a probabilistic graphical model that factors into three key components. The first component grounds landmark phrases such as "the computers" in the perceptual frame of the robot by exploiting co-occurrence statistics from a database of tagged images such as Flickr. Second, a spatial reasoning component judges how well spatial relations such as "past the computers" describe a path. Finally, verb phrases such as "turn right" are modeled according to the amount of change in orientation in the path. Our system follows 60% of the directions in our corpus to within 15 meters of the true destination, significantly outperforming other approaches.**

***Index Terms*—spatial language, direction understanding, route instructions**

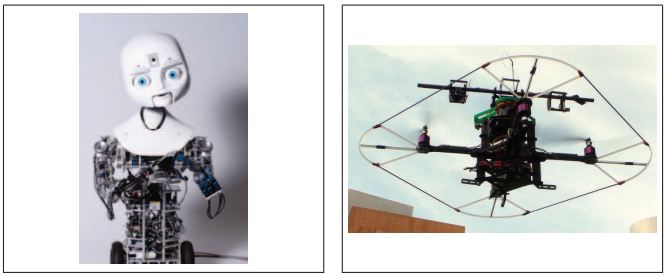
I. Introduction

Natural language is an intuitive and flexible modality for human-robot interaction. A robot designed to interact naturally with humans must be able to understand instructions without requiring the person to speak in any special way. Understanding language from an untrained user is challenging because we are not asking the human to adapt to the limitations of the system, i.e., to limit their instructions to a small vocabulary or grammar. Rather, we want a system that understands naturalistic language directly as produced by people.

Our work is directed toward understanding naturalistic language as part of a larger multi-university effort to develop autonomous robot teammates that collaborate naturally and effectively with humans in a civilian response after a mass casualty event. Two types of robots take part in the scenario: quadrotor helicopters and the Mobile Dexterous Social robot (MDS), an expressive, mobile humanoid (Figure 1(a)). In the first-responder scenario, robots will need to interact with both trained and untrained humans to aid the rescue effort. For example, a robot might engage in the following dialog when it finds a human victim:

- **Robot** Someone is on the way to get you out of here. Are there any other people around who need help?
- **Victim** I saw someone in the main lobby.
- **Robot** Where is the main lobby?

*The first two authors contributed equally to this paper.



(a) humanoid (b) helicopter

Fig. 1. Robot platforms

With your back to the windows, walk straight through the door near the elevators. Continue to walk straight, going through one door until you come to an intersection just past a white board. Turn left, turn right, and enter the second door on your right (sign says "Administrative Assistant").

Fig. 2. A typical set of directions from the corpus described in Section III.

- **Person** Exit this room and turn right. Go down the hallway past the elevators. The lobby is straight ahead.
- **Robot** Understood.

In such scenarios, language has a strong advantage over other interaction modalities because humans do not need special training to interact with the robot, and can keep their hands and eyes free for other tasks. However building robust language understanding systems that can actually engage in dialog in realistic situations remains a challenging problem. The management of this interaction requires at least four abilities: speech recognition, dialogue management, state estimation (e.g. parsing and grounding of the symbols in the percepts of the robot) and planning, any one of which could fail.

We are taking steps towards building a robust natural language system by focusing on understanding a subset of the tasks in the first-responder scenario, that of direction understanding (see Figure 2). We focus on direction understanding for several reasons. First, following directions requires the ability to understand spatial language. Because spatial language is pervasive, this ability is important for almost any application of natural language to robotics. Second, a system that understands directions is useful in many other scenarios, including health care and companion robots. Third, it is natural to ask humans to create a set of directions through an environment, yielding an open-ended yet task-constrained corpus of language. Finally, there is a natural correctness metric when evaluating a robot's performance at following natural language directions: did it reach the correct final destination? The availability of a corpus and a concrete correctness metric enable an offline component-based evaluation of our

system, which is critical for achieving robustness, because we can quickly test new models on linguistic input from many different users.

Given the scenario of direction understanding, our goal is to create a system that will take as input a direction, as in Figure 2, and infer the intended path through the environment. We do this by extracting shallow linguistic structure from the directions, grounding elements from that structure in the environment, and performing inference to find the most probable path through the environment given the directions and observations. The text of the directions is first parsed to a sequence of *spatial description clauses* (SDCs). Our system then takes the sequence of SDCs, a partial or complete semantic map of the environment, and a starting location, and it outputs a sequence of waypoints through the environment, ending at the destination. Planning a path is formulated as finding the maximum probability sequence of locations in a map. This inference uses a probabilistic graphical model that is factored into three key components. The first component grounds novel noun phrases such as "the computers" in the perceptual frame of the robot by exploiting object co-occurrence statistics between unknown noun phrases and known perceptual features. These statistics are learned from a large database of tagged images such as Flickr. Second, a spatial reasoning component judges how well spatial relations such as "past the computers" describe a path. Third, verb phrases such as "turn right" are modeled according to the amount of change in orientation in the path. Once trained, our model requires only a grid-map of the environment together with the locations of detected objects in order to follow directions through it. This map can be given *a priori* or created on the fly as the robot explores the environment.

At this stage of our work, we are focusing more on the technical feasibility of our approach at the specific subtask of direction giving, rather than on the overall usability of a fully-functional natural language interface. To evaluate the technical feasibility, we collected a corpus of 150 natural language route instructions from fifteen people, through one floor of two adjoining office buildings. An example set of directions from the corpus is shown in Figure 2. Following these directions is challenging because they consist of natural language constrained only by the task and as a result may use any of the complicated linguistic structures associated with free-form natural language. The highest performing model searches the entire semantic map to successfully follow 60% of the directions in our corpus, significantly outperforming a baseline that uses only landmark phrases in the directions. We also tested our approach with an exploration-based algorithm that does not have a map of the environment *a priori*, showing that spatial relations improve performance in unknown environments.

II. Related Work

Many authors have proposed formalisms similar to spatial description clauses for enabling systems to reason about the semantics of natural language directions. For example, Bugmann et al. [1] identified a set of 15 primitive procedures associated with clauses in a corpus of spoken natural language directions. Our work follows their methodology of corpus-based robotics but focuses on achieving robust understanding of natural language directions rather than an end-to-end system. Levit and Roy [2] designed *navigational informational units* that break down instructions into components. MacMahon et al. [3] represented a clause in a set of directions as a compound action consisting of a simple action (move, turn, verify, and declare-goal), plus a set of pre- and post-conditions. Our local search algorithm most closely corresponds to this system. Look et al. [4] created an ontology for modeling the relationships between spaces, and used it for generating natural language directions. Many of these representations are more expressive than SDCs but are more difficult to automatically extract from text; many authors sidestep this problem by using human annotations (e.g., [2, 3]).

Others have created language understanding systems that follow natural language commands, but without using a corpus-based evaluation to enable untrained users to interact with the system (e.g., [5, 6]). Bauer et al. [7] built a robot that can find its way through an urban environment by interacting with pedestrians using a touch screen and gesture recognition system.

The structure of the spatial description clause builds on the work of Landau and Jackendoff [8], and Talmy [9], providing a computational instantiation of their formalisms. We are currently drawing from Levin [10] to develop a richer model of verbs, including ditransitive verbs such as "bring." Many of the features used to model the semantics of spatial prepositions are directly inspired from their work. Building on the paradigm of testing the semantics of spatial prepositions against human judgments [11, 12], this work applies the models to understanding natural language directions in realistic environments.

In previous work, we have built direction understanding systems that model the directions as a sequence of landmarks [13]. This work builds on the previous system by introducing the SDC formalism and adding models for spatial relations and verbs, enabling system to more completely model the natural language, while still exploiting the structure of landmarks in the map to follow the directions.

III. Natural Language Directions

To understand the language used to give directions, and to evaluate the technical feasibility of our system, we collected a corpus of natural language directions through an office environment in two adjoining buildings at MIT. Our goal in collecting this corpus was to evaluate the accuracy of our system at following natural language directions rather than to perform an end-to-end usability evaluation. We asked fifteen subjects to write directions between 10 different starting and ending locations, for a total of 150 directions. Subjects were solicited by placing flyers around MIT and were selected for inclusion in the study if they were between the ages of 18 and 30 years old, were proficient in English, and were unfamiliar with the test environment. The pool was made up of 47% female and 53% male subjects from the MIT community, primarily students or administrators.

When collecting directions, we first gave subjects a tour of the building to familiarize them with the environment. Then we asked subjects to write down directions from one location in the space to another, as if they were directing a friend. Subjects were allowed to wander around the floor as they wrote the directions and were not told that this data was for

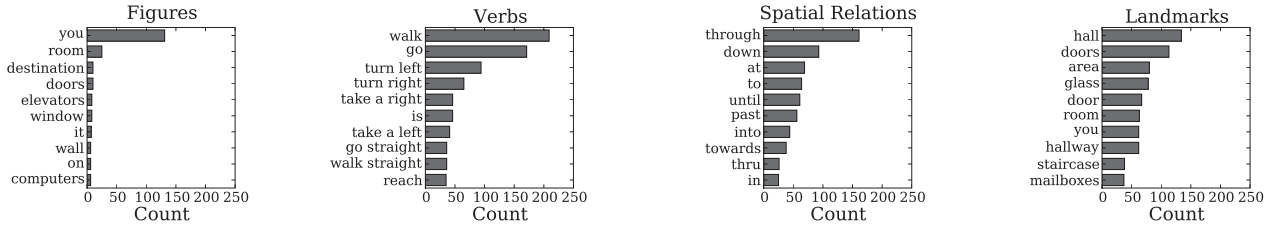


Fig. 3. Histogram showing the most frequent words that appear in each of the fields of an SDC from our corpus. For Figures and Landmarks, similar phrases have been grouped together.

a robotics research experiment. Experimenters did not refer to any of the areas by name, instead using codes labeled on a map. After writing these directions, the subjects were asked to follow a different set of directions that another subject had created, in order to understand if there were certain people who were better or worse at giving point-to-point directions. The experimenter would read the directions to the subject and follow the subject as the subject followed directions. When the subject became lost, they were asked to report this and the trial was concluded. Subjects successfully followed 85% of the directions in the corpus used in our evaluation. 100% of the directions of the best direction giver were successfully followed, while only 30% of the directions from the worst direction giver could be followed.

In order to enable an offline evaluation, our corpus included a log of a robot's observations of the environment. To collect the dataset, we used a mobile robot instrumented with a LIDAR and camera. The log thus contains odometry, laser scans, and camera images. The laser scans and odometry were used by a SLAM module in order to create a map as the robot explores the environment [14], while the camera was used to detect baseline objects in order to create a semantic map. The log enables us to test our models at following directions from the corpus offline, without the overhead of deploying a physical robot in a user study.

IV. Spatial Description Clauses

In order to follow the directions in our corpus, the system exploits the structure of the language in the directions. First, directions are sequential: in most cases each phrase in the directions refers to the next region in the environment on the way to the final destination. Second, directions contain references to landmarks that the person is meant to see along the path to the destination region. Next, spatial relations such as "past" and "through" describe how a person should move relative to these landmarks. Finally, directions contain imperative verbs which tell a person what to do and where to go. The hierarchical structure of language relates all these components together.

We formalized this structure by modeling each sentence in a set of directions as a hierarchy of structured clauses. Each spatial description clause (SDC) consists of a *figure* (the subject of the sentence), a *verb* (an action to take), a *landmark* (an object in the environment), and a *spatial relation* (a geometric relation between the landmark and the figure). Any of these fields can be unlexicalized and therefore only specified implicitly. For example, in the sentence "Go down the hallway," the figure is an implicit "you," the verb is "go," the spatial relation is "down" and the landmark is "the hallway." SDCs are also hierarchical. For the sentence "Go through the set of double doors by the red couches," the top level SDC has a verb, "go," a spatial relation, "through," and a landmark, "the set of double doors by the red couches," while the landmark contains a nested SDC with figure "the set of double doors," spatial relation "by" and landmark "the red couches." Figure 4(a) shows the hierarchy of SDCs for a sentence in our corpus.

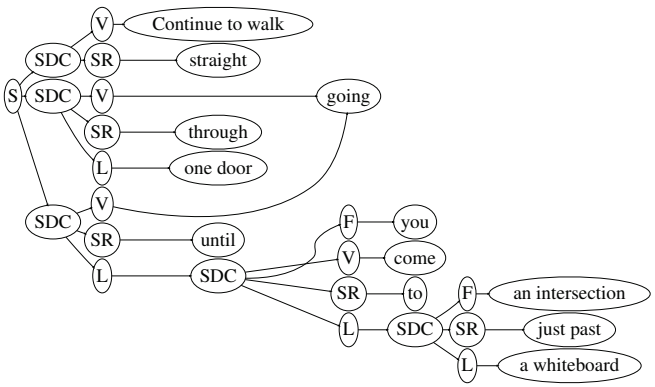
We hand-annotated the text of 150 directions in our corpus with the SDCs in order to verify that SDCs are capable of capturing the linguistically expressed structure of the directions. Nearly all of the sentences in the dataset can be parsed into SDCs that correctly preserve the semantics of each word in the sentence, with very few (7.29%) orphaned words, virtually all stop words. Figure 3 shows the top ten words that appeared in each field of an SDC in our corpus. Some types of sentences could not be parsed into the SDC formalism, such as commands not to do something, multi-argument verbs, and ambiguous prepositional phrase attachment (e.g., "Follow the wall to the small kitchen"). These limitations of SDCs could be addressed with a more complex framework. However, SDCs capture important parts of the semantics of route instructions in our corpus, and they are efficient to extract and use in inference.

V. System

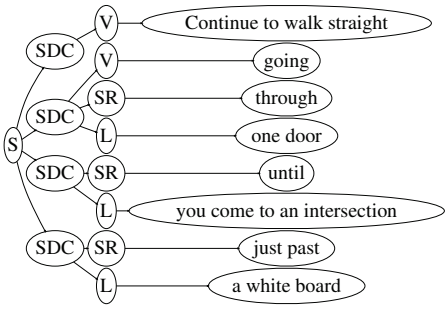
Our system uses SDCs to follow natural language directions by finding a path that maximizes the joint distribution of paths and SDCs, given detected objects. In order to implement this model, we must automatically extract SDCs from the text, and then ground each of the parts of the SDC in the environment.

A. Automatically Extracting Spatial Description Clauses

We designed SDCs to capture the hierarchical structure of directions since this structure seems important when grounding landmarks. However, to make our model tractable, we approximated the hierarchical structure of SDCs as a sequence. A conditional random field (CRF) model automatically extracts SDCs from text [15]. The CRF labels each word in each sentence with one of the four possible fields (*figure*, *verb*, *spatial relation* and *landmark*), or none. (The CRF was trained on a different corpus of route instructions from the one used in our evaluation.) A greedy algorithm groups continuous chunks together into SDCs. Figure 4(b) shows the SDCs generated by this component for one sentence. Although it lacks the hierarchical structure of the annotated data (as in Figure 4(a)), the SDCs capture the sequential structure of the directions, and segments the key components of each phrase. Quantitatively, 60% of the SDCs produced by the CRF correspond exactly to an SDC in the hand-annotated



(a) Ground Truth



(b) Automatic

Fig. 4. Ground-truth and automatically extracted SDCs for the sentence, "Continue to walk straight, going through one door until you come to an intersection just past a white board." Here, *S* is the entire sentence, *SDC* is a spatial description clause, *F* is the figure, *V* is the verb, *SR* is the spatial relation, and *L* is a landmark.

ground truth created by one of the authors. To measure inter-annotator agreement, a second person annotated the SDCs in our corpus, and also had 60% agreement with one of the authors. When the automatic algorithm makes a mistake, it is usually a boundary problem, for example including the spatial relation and landmark, but excluding the verb. In these cases, the annotations still contain structured information that can be used to follow the directions.

B. Topological Map

Rather than searching all possible paths through the environment, the system first creates a topological roadmap from the gridmap of the environment and searches for a path within this graph. The roadmap is created by automatically segmenting spaces based on visibility and detected objects and then extracting a topology of the environment from this segmentation, building on techniques described by Brunskill et al. [16]. Figure 5 shows a floorplan of the environment used in our corpus, together with the nodes and edges in this roadmap. Each starred location in Figure 5 contains four viewpoints v_i facing in each of the four cardinal directions.

C. Model

We formulate the problem of understanding natural language directions as inferring a sequence of viewpoints given a set of natural language directions. When P is the path, S is the sequence of the SDCs, and O are the detected objects, we want to compute:

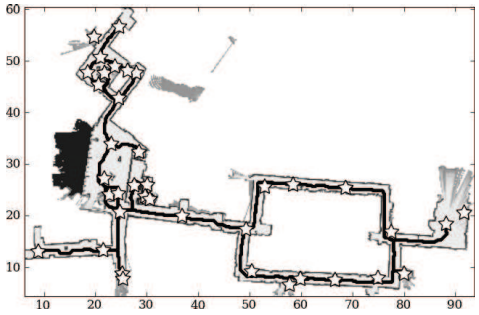


Fig. 5. A map of the environment used to collect our corpus of natural language directions, together with the automatically extracted roadmap. Each star is a node in the topological map. (Distances are in meters.)

$$\arg\max_{\mathrm{P}} p(P, S|O) = p(S|P, O) \times p(P|O) \tag{1}$$

We model paths as transitions between viewpoints in the environment v_i, and the directions as a list of SDCs. We also assume the path is independent of the detected objects, leading to:

$$p(P, S|O) \approx p(\mathrm{sdc}_1 \ldots \mathrm{sdc}_M | v_1 \ldots v_{M+1}, O) \times p(v_1 \ldots v_{M+1})$$

We can factor this distribution into a component for the path and a component for the observed SDCs. In particular, we assume that an SDC depends only on the current transition v_i, v_{i+1}, and that the next viewpoint v_{i+1} depends only on previous viewpoints. These two assumptions lead to the following factorization:

$$p(P, S|O) = \left[\prod_{i=1}^{M} p(\mathrm{sdc}_i | v_i, v_{i+1}, O)\right] \times \left[\prod_{i=1}^{M} p(v_{i+1} | v_i \ldots v_1)\right] \times p(v_1) \tag{2}$$

The most important part of our model is the observation probability, $p(\mathrm{sdc}_i | v_i, v_{i+1}, O)$. To compute this probability, we break down the SDC into its component parts: the figure, f, the verb or action, a, the spatial relation, s, and the landmark, l. Given that v_i is the ith viewpoint and o_k is the ith detected object, we can obtain the following distribution:

$$\begin{aligned} p(\mathrm{sdc}_i | v_i, v_{i+1}, O) &= p(f_i, a_i, s_i, l_i, | v_i, v_{i+1}, O) \\ &\approx p(f_i | v_i, v_{i+1}, o_1 \ldots o_K) \times p(a_i | v_i, v_{i+1}) \times \\ &\quad p(s_i | l_i, v_i, v_{i+1}, o_1 \ldots o_K) \times p(l_i | v_i, v_{i+1}, o_1 \ldots o_K) \end{aligned} \tag{3}$$

At this point, we have factored the distribution into four parts, corresponding to each field of the SDC, plus transition probabilities. We model the transition probabilities in the second term of equation 2 as uniform among connected viewpoints in the topological map, together with a constraint that disallows backtracking. This constraint means that the path is not allowed to revisit any location that it has previously visited. The following sections describe the other terms.

D. Grounding the verb field

The verb/action field models verbs in one of three ways: "turn left," "turn right," and "go straight." The type of the verb is computed based on keywords in the verb field of the

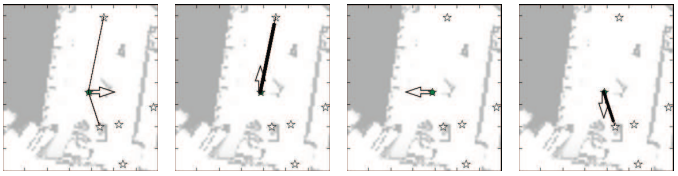


Fig. 6. Transitions for "straight," in four different orientations. The thickness of the line corresponds to the probability of the transition. There is a high probability of transition to viewpoints directly in front of the current location, and a lower probability of transition to viewpoints to one side or the other.

SDC; the default type is "straight." The probability of the verb is computed according to the total amount of change in orientation required to travel between two viewpoints:

$$p(a_i|v_i, v_{i+1}) \approx \max(1 - \frac{C}{\pi}[\text{turn}(v_i, v_{i+1}) - \theta], 0) \quad (4)$$

We assume natural robot motion: in order to move from one viewpoint to another the robot must first turn to the destination, drive there, and then turn to its final orientation. The total turn amount corresponds to how much the robot must turn in order to achieve this. For "left," θ is $90°$; for "right" it is $-90°$ while C is 2.5. For "straight," θ is 0 and C is 1.75. θ sets the desired orientation changed associated with the verb, while C sets how much this error penalizes the transition. Figure 6 shows a visualization of these values. In the future we plan to expand our model of verbs to distinguish between directives to move, directives to change orientation, and descriptions of expected landmarks in the environment.

E. Grounding the figure and landmark fields

The system models the likelihood of the landmarks in an SDC given a viewpoint transition and detected objects. This problem is challenging because people refer to a wide variety of objects in natural language directions, and use diverse expressions to describe them. In our corpus people utilized more than 150 types of objects as landmarks, ranging from "the door near the elevators" to "a beautiful view of the domes." (Figure 3 shows the most frequent landmarks.) To ground landmark phrases, the system takes a semantic map seeded with the locations of 21 types of known objects, and uses object-object context to predict the locations of the unknown landmark terms, following Kollar and Roy [17]. Object-object context allows the system to predict that a computer is nearby if it can directly detect a monitor and a keyboard.

To predict where a novel landmark may occur, we downloaded over a million images, along with their associated labels. We used the photo-sharing website Flickr to accomplish this, although any dataset where the images and co-occurrence counts were available could have been used. Using the co-occurrence counts we computed the probability of seeing a novel landmark l_i given the detected objects o_k:

$$p(l_i|v_i, v_{i+1}, o_1 \dots o_K) = p(l_i|\text{visible_objects}(v_i)) \quad (5)$$

$$\approx \max_{o_k \in \text{visible_objects}(v_i)} p(l_i|o_k) \quad (6)$$

We use the maximization heuristic as a way to smooth the distribution because detecting additional landmarks in the environment usually only increases the probability of seeing the landmark phrase from the directions. For example, $p(\text{kitchen}|\text{microwave}, \text{toaster})$ should always be larger than $p(\text{kitchen}|\text{microwave})$. This heuristic tries to pick the subset of visible objects that minimizes the entropy of the distribution. When there is a noun phrase in the *figure* field besides "you," we model it in the same way as the landmark field.

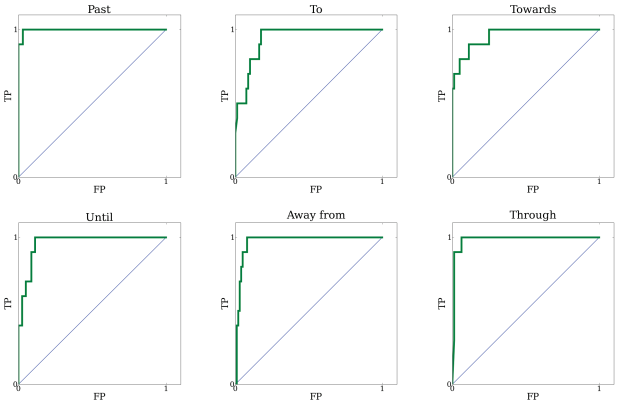


Fig. 7. When treating the spatial relation probability as a classifier, above is the performance on the corpus of examples drawn for a particular path. On the horizontal axis of the ROC curve is the false positive rate (FP) and on the vertical axis is the true positive rate (TP).

F. Grounding Spatial Relations

To use spatial relations to follow directions in our model, we need to compute how well a phrase such as "past the door" describes a particular path segment, $[v_i, v_{i+1}]$. Here we focus on dynamic spatial prepositions that describe the properties of a path, as opposed to static spatial prepositions that localize an object, since almost all of the most frequent spatial prepositions in our corpus describe a path (Figure 3). We conceive of spatial relations as two-argument functions that take a figure and a landmark. For dynamic spatial prepositions, the figure is a path, represented as a sequence of points, and the landmark is a point or a polygon. We want to compute the following, where s_i is ith spatial relation, l_i is the ith landmark, and o_k are the detected objects, which consist of a location and a bounding polygon.

$$p(s_i = \text{past}|l_i = \text{door}, v_i, v_{i+1}, o_1 \dots o_K) = \sum_{o_k} p(s_i = \text{past}|\text{landmark} = o_k, \text{path} = v_i, v_{i+1}) \times p(l_i = \text{door}|\text{visible_objects}(o_k)) \quad (7)$$

We sum over all possible landmark locations because the system does not know which physical door is referred to by the phrase "the door." This marginalization causes the system to prefer paths that pass many doors to those that pass only a few. By taking the path that goes past many doors, the system is more likely to pass the one referred to in the directions.

To model the first term in equation 7, we introduce features that capture the semantics of spatial prepositions. These features are functions of the geometry of the path and landmark. For example, one of the features utilized for the spatial preposition "to" is the distance between the end of the path and the landmark's location. We used Naive Bayes to model the distribution $p(s_i = \text{past}|\text{landmark} = o_i, \text{path} = v_i, v_{i+1})$. Features are described in detail in Tellex and Roy [18].

To integrate these features into our model, our system

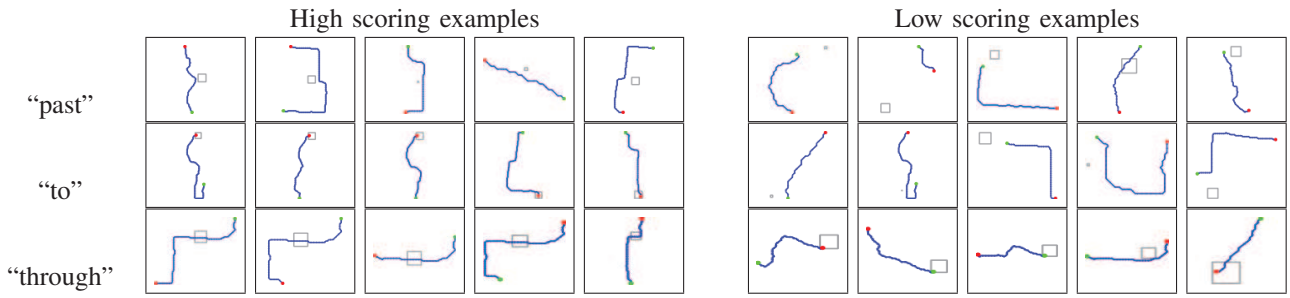


Fig. 8. Five high scoring and five low scoring examples that were found in our data set for several spatial prepositions.

learns the distribution in equation 7 from a dataset created by hand-drawing examples of paths that matched a natural language description such as “through the door.” In this data set, positive examples of one spatial relation were taken to be negative examples of others. Some pairs of spatial relations, such as “to” and “towards,” which are very similar, were excluded from each other’s training sets. This dataset was collected from a different training environment and generalizes across environments. If we treat the resulting distribution as a classifier for a particular spatial relation, then the performance on a held-out test set from this corpus is shown in Figure 7. Some of the highest and lowest scoring examples are shown in Figure 8. We trained classifiers for eleven spatial prepositions: “across”, “along,” “through,” “past,” “around,” “to,” “out,” “towards,” “down,”, “away from,” and “until.” (“Until” is not a spatial relation in general, but we modeled it as one here because it almost always refers to an arrival event in our corpus, as in “until you come to an intersection just past a whiteboard.”)

G. Performing Inference

Once the model is trained, our system can infer paths through any environment. If the robot has explored the entire area *a priori* and has access to a map of the environment, *global* inference searches through all possible paths to find the global maximum of the joint distribution. When a full map is unavailable, the robot uses a greedy *local* inference algorithm that searches for paths using only local information. We perform global inference using a Viterbi-style algorithm [19] that finds the most probable sequence of viewpoints corresponding to a given sequence of SDCs. The algorithm takes as input a starting viewpoint, a map of the environment with some labeled objects, and the sequence of SDCs extracted from the directions. It outputs a series of viewpoints through the environment, using the model described above to compute the probability of a transition between two viewpoints.

The local inference algorithm iterates over the SDCs and at each step chooses the next viewpoint v_{i+1} that maximizes $p(sdc_i|v_{i+1}, v_i, o_1 \dots o_K) * p(sdc_{i+1}|v_{i+1}, v_{i+2}, o_1 \dots o_K) * p(v_{i+1}|v_i) * p(v_{i+2}|v_{i+1})$. In other words, it looks ahead two SDCS, and chooses the best transition from among the children and grandchildren of the current node. We expect global inference to perform better because it searches through all possible paths to find the one that best matches the descriptions. However, the local inference is more practical for a real robot, because it does not require the robot to have built a complete map of the environment and objects in it before following directions.

TABLE I
THE PERFORMANCE OF OUR MODELS AT 10 METERS.

Algorithm	% correct	
	Max Prob	Best Path
Global inference w/spatial relations	48.0%	59.3%
Global inference w/o spatial relations	48.0%	54.7%
Local inference w/ spatial relations	28.0%	42.0%
Local inference w/o spatial relations	26.7%	30.7%
Wei et al. [13]	34.0%	34.0%
Last SDC only	23.0%	24.0%
Random	0.0%	–

VI. EVALUATION

To evaluate the technical feasibility of our approach, we performed a component-level evaluation of our system, measuring its performance at following natural language directions from our corpus. For each set of directions, the system tried all four possible starting orientations. We chose two evaluation metrics. For the maximum probability metric, only the highest probability path from the four starting orientations is evaluated. In the best-path metric, only the path that ended up closest to the true destination is evaluated. We chose the latter metric because the true starting orientation of the subject at the beginning of each set of directions was difficult to automatically determine. Figure 9 shows a comparison of our model to three baselines: on the horizontal axis is the distance from the final location of the inferred path to the correct destination, on the vertical axis is the percentage correct at that distance. Performance differences at 10 meters are shown in Table I. We present performance at a threshold qualitatively close to the final destination in order to compare to human performance on this dataset, which is 85%.

The first baseline (*Random*) is the expected distance between the true destination and a randomly selected viewpoint. The second (*Last SDC*) returns the location that best matches the last SDC in the directions. The third baseline (*Landmarks Only*) corresponds to the method described by Wei et al. [13], which performs global inference using landmarks visible from any orientation in a region, and no spatial relations or verbs. Our global inference model significantly outperforms these baselines, while the local inference model slightly outperforms Wei et al. [13] despite not performing global search.

We were especially interested in the performance of our model with and without spatial relations, since they are a key difference between our model and previous work. Figure 10 shows the performance of these models for all subjects, while Figure 11 shows the performance for the

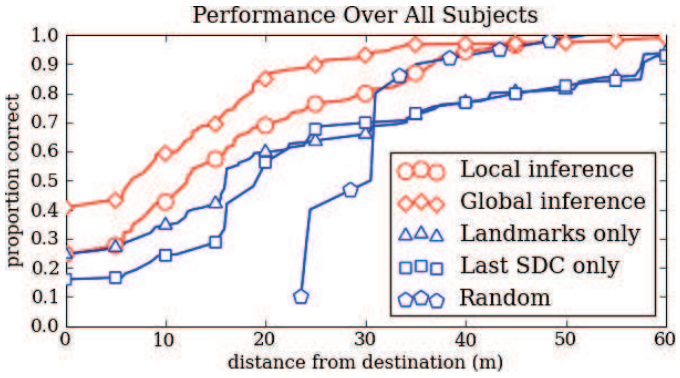


Fig. 9. Comparison of our model to baselines. Versions of our model are shown in red; baselines are in blue.

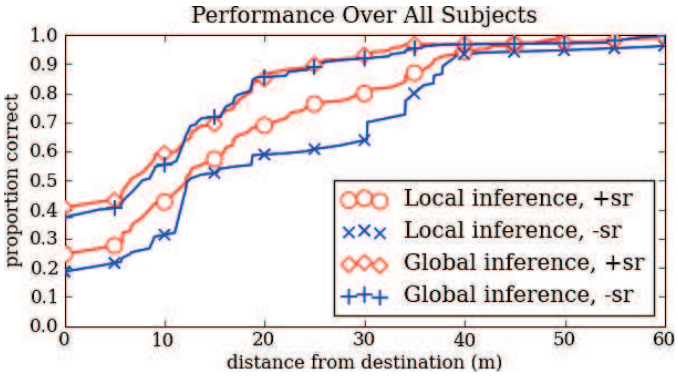


Fig. 10. Comparison of global inference algorithm with local inference, with and without spatial relations, using the best-path metric.

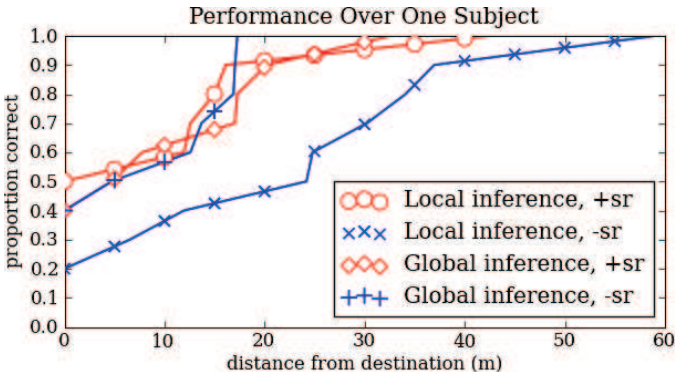
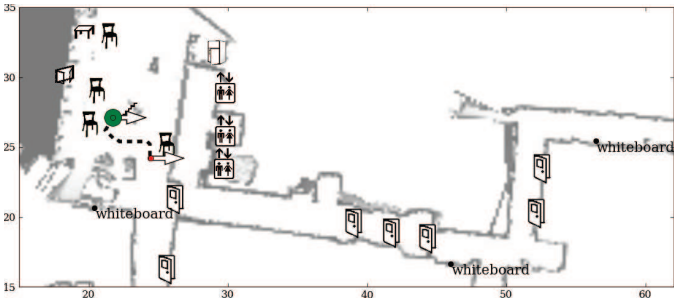
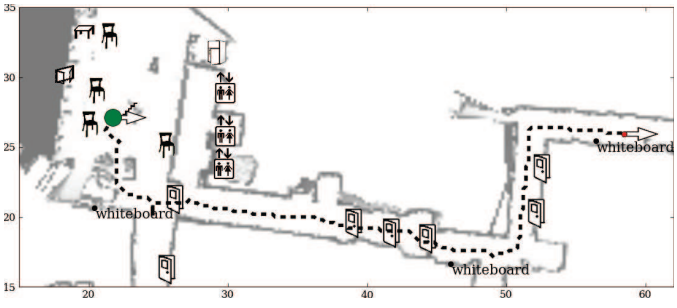


Fig. 11. Comparison of the global inference algorithm with the greedy approach, with and without spatial relations for directions from one of the best-performing subjects, using the best-path evaluation metric.



(a) Without spatial relations.



(b) With spatial relations.

Fig. 12. Paths outputted by the local inference system with and without spatial relations. The path without spatial relations is extremely short and completely incorrect.

subject whose directions had the highest performance with the global inference algorithm with spatial relations. Spatial relations do not contribute much to the performance of the global inference algorithm, but do increase the performance of the local inference algorithm. For one of our subjects, they raise the performance of the local search algorithm into the range of the global inference algorithms. Possibly this effect is because when deciding to go through a particular door, the global inference algorithm searches on the other side of that door, and if landmarks farther along in the sequence of SDCs match that path, then it will go through the door anyway. In contrast, the local search approach benefits more from spatial relations because it cannot see the other side of the door, so relying on the geometric features of the path helps to disambiguate where it should go. Figure 12(a) and Figure 12(b) show the paths inferred by the two models for the set of directions from Figure 2. Without spatial relations, the model is content to stay in the first room, from which it can see objects, such as a whiteboard and a door, that occur in the directions. In contrast, the model that uses spatial-relations goes through the first door when told to "walk straight through the door near the elevators" and ends up at the correct final destination. This result suggests that the role of spatial relations in natural language directions is to help the direction follower disambiguate these local decisions, so that they do not have to perform a full search of the environment in order to follow the directions.

The most significant improvement in performance over the system corresponding to Wei et al. [13] comes from the model of verbs with viewpoints as described in Section V-D, suggesting that the combination of verbs and landmarks is critical for understanding natural language directions. We were surprised that a relatively simple model of verbs, involving only left, right, and straight, caused such a large improvement compared to the effect of spatial relations.

VII. Conclusion

In this work, we have presented a system that understands task-constrained natural language. Although the results we have shown are promising, they are not yet definitive. Because of parser failures, access to only 21 known object types, and few distinguishing features at some destinations, we expect the system's performance to be below that of humans. In doing

a failure analysis, many directions could not be followed because they used landmarks which the system failed to resolve. For example, subjects often referred to places where the carpet changed color. Incorporating models of color adjectives might address some of these problems. Also, our model of spatial relations sees landmarks only as points. As a result, sentences like "Go down the hallway" probably did not add much information to the inference because the system does not know the actual geometry of the hallway. Modeling the expected size of a landmark phrase could address this issue. Although local inference is important for understanding directions in new environments, our local inference algorithm did not perform as well as global inference. To fix this problem, we are developing algorithms that explore the environment, build a semantic map on the fly, and backtrack to try another route if an error is detected. Our model already achieves a significant fraction of human performance, and by exploiting more linguistic information from the directions and contextual information from other large corpora, we hope to make our system even more robust. We have shown the technical feasibility of our approach; in the future we intend to show that it generalizes to other environments and evaluate its usability as part of a complete natural language understanding system.[1]

Robustly following natural language directions is only part of a complete natural language interface. A complete system requires the ability to understand many more commands in different scenarios, the ability to engage in dialogue with people, and, for many applications, speech recognition. While many are focusing on the latter two aspects, we are currently investigating a richer model that will allow us to understand natural language queries in more general scenarios, such as "Is Daniela in her office?" or "Wait for John at the elevators. When he arrives, bring him here." Our approach is to develop a corpus of commands that people use in natural situations, find models for the meanings of the most relevant words, then formulate a probabilistic model to compose the meanings together to infer a plan for the robot.

In this work, we have demonstrated an approach to enable a robot to infer a plan from natural language directions. We developed spatial description clauses, a formalism that captures the semantics of natural language directions. Our system automatically extracts SDCs from natural language input and uses them to find a path through the environment corresponding to the directions. In order to connect SDCs to the environment, our system builds a semantic map, that contains some detected landmarks, and then utilizes co-occurrence statistics from Flickr to predict the locations of objects referred to in the directions. It uses models of the meanings of spatial prepositions to understand spatial relations that appear in directions. A probabilistic model connects these pieces together, finding the path through the environment that maximizes the probability of the directions. Our system takes as input free-form natural language directions and infers paths through real environments.

VIII. Acknowledgments

We are grateful for the support of the Office of Naval Research, which supported Thomas Kollar and Stefanie Tellex under MURI N00014-07-1-0749. We would like to thank our HRI reviewers, Finale Doshi, Jeff Orkin, Rony Kubat, Olivier Koch, and Cynthia Breazeal for their comments on drafts of this paper, and Sachi Hemachandra and Emma Brunskill for their help in collecting the corpus.

[1]More information about our work is available at http://du.tkollar.com.

References

[1] G. Bugmann, E. Klein, S. Lauria, and T. Kyriacou, "Corpus-based robotics: A route instruction example," *Proceedings of Intelligent Autonomous Systems*, pp. 96—103, 2004.

[2] M. Levit and D. Roy, "Interpretation of spatial language in a map navigation task," *Systems, Man, and Cybernetics, Part B, IEEE Transactions on*, vol. 37, no. 3, pp. 667–679, 2007.

[3] M. MacMahon, B. Stankiewicz, and B. Kuipers, "Walk the talk: Connecting language, knowledge, and action in route instructions," *Proceedings of the National Conference on Artificial Intelligence*, pp. 1475—1482, 2006.

[4] G. Look, B. Kottahachchi, R. Laddaga, and H. Shrobe, "A location representation for generating descriptive walking directions," in *International Conference on Intelligent User Interfaces*, 2005, pp. 122–129.

[5] 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 *IEEE International Conference on Robotics and Automation*, 2009, pp. 4163–4168.

[6] M. Skubic, D. Perzanowski, S. Blisard, A. Schultz, W. Adams, M. Bugajska, and D. Brock, "Spatial language for human-robot dialogs," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 34, no. 2, pp. 154–167, 2004.

[7] A. Bauer, K. Klasing, G. Lidoris, Q. Mhlbauer, F. Rohrmller, S. Sosnowski, T. Xu, K. Khnlenz, D. Wollherr, and M. Buss, "The Autonomous City Explorer: Towards natural human-robot interaction in urban environments," *International Journal of Social Robotics*, vol. 1, no. 2, pp. 127–140, Apr. 2009.

[8] B. Landau and R. Jackendoff, ""What" and "where" in spatial language and spatial cognition," *Behavioral and Brain Sciences*, vol. 16, pp. 217–265, 1993.

[9] L. Talmy, "The fundamental system of spatial schemas in language," in *From Perception to Meaning: Image Schemas in Cognitive Linguistics*, B. Hamp, Ed. Mouton de Gruyter, 2005.

[10] B. Levin, *English Verb Classes and Alternations: A Preliminary Investigation*. University Of Chicago Press, Sep. 1993.

[11] T. P. Regier, "The acquisition of lexical semantics for spatial terms: A connectionist model of perceptual categorization," Ph.D. dissertation, University of California at Berkeley, 1992.

[12] J. D. Kelleher and F. J. Costello, "Applying computational models of spatial prepositions to visually situated dialog," *Computational Linguistics*, vol. 35, no. 2, pp. 271–306, Jun. 2009.

[13] Y. Wei, E. Brunskill, T. Kollar, and N. Roy, "Where to go: Interpreting natural directions using global inference," in *IEEE International Conference on Robotics and Automation*, 2009.

[14] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with Rao-Blackwellized particle filters," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 34–46, 2007.

[15] T. Kudo, "CRF++: Yet another CRF toolkit," http://crfpp.sourceforge.net, 2009.

[16] E. Brunskill, T. Kollar, and N. Roy, "Topological mapping using spectral clustering and classification," in *International Conference on Intelligent Robots and Systems*, October 2007, pp. 3491–3496.

[17] T. Kollar and N. Roy, "Utilizing object-object and object-scene context when planning to find things." in *IEEE International Conference on Robotics and Automation*, 2009.

[18] S. Tellex and D. Roy, "Grounding spatial prepositions for video search," in *Proceedings of the International Conference on Multimodal Interfaces*, 2009.

[19] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *Information Theory, IEEE Transactions on*, vol. 13, no. 2, pp. 260–269, 1967.