A Framework for Understanding Verbal Route Instructions

Matt MacMahon
adastra@mail.utexas.edu
Department of Electrical and Computer Engineering
Intelligent Robotics Laboratory
University of Texas at Austin

Abstract

My goal is to understand human verbal route instructions by modeling and implementing the language, knowledge representation, and cognitive processes needed to communicate about spatial routes. To understand human route instructions, I ran a study of how people give and follow route instructions. I modeled the language used in the route instruction texts using standard computational linguistics techniques. I model the information content of the route instruction texts using a spatial ontology. I introduce a set of axioms and heuristics that can autonomously transform a linguistic representation of what the route director said into a semantic representation of the intended route. This semantic representation has been implemented in mobile robots that can navigate and accomplish simultaneous localization and mapping in real-world environments. Finally, I model the processes of following and giving route instructions as Markov processes.

Problem definition

A *route instruction* is an instruction intended to guide a mobile agent toward a spatial destination. A *route instruction set* is the collection of route instructions that describes a route. A route instruction set is also referred to as "route directions" or "directions" [1]. Route instructions are a special case of verbal instructions, which include recipes, assembly instructions, and usage manuals. While route instruction sets may include verbal, gestural, and pictorial components, (respectively using words, body gestures, and map elements) this work focuses on verbal route instructions. The route instruction provider is referred to here as the *director* and the agent following a route instruction set as the *follower*.

This paper presents a framework to develop systems that communicate route instructions in natural language. As a route instruction follower, the system represents verbal route descriptions in a spatial knowledge ontology that a robot can use to navigate the route. As a director, the system generates route instructions conveying similar information in similar styles as people's route instructions.

I modeled the language of the route instruction texts, using a learned probabilistic context-free grammar to parse the sentences. These parse trees were converted into attribute-value matrices to filter out the incidental features of the sentences, such as phrase order. I combined the linguistic representations of route instruction texts into a representation of the intended route map, in terms of the topological layer of the *Spatial Semantic Hierarchy*. I introduce a family of Markov processes that model route directors and followers of various abilities and goals.

Other work on route instructions

Route instruction understanding

Several software systems analyze or follow route instructions. Riesbeck's system evaluated route instructions by high-level characteristics, independent of the environment (1980). His natural language parsing and understanding program analyzed a set of route instructions for overall *clarity* and *cruciality* measures. Each motion must be described completely and precisely (clarity); additional descriptions provide checks but are not crucial.

Webber *et al.* looked at the broader question of inferring an intended plan from any instructions (1995). Müller *et al.* implemented a system which can follow a formal route description through an environment, with the intention of adding on a natural language understanding system (2000).

Perzanowski *et al.* combined a speech recognizer, a deep parser and a dialog model with hand gesture recognition, and other deictic references on a Palm Pilot (2001). This work was part of GRACE, a robot system that navigated through a conference center by asking for and following route instructions (Simmons & others 2003). Frank suggested formalizing verbal route instructions into action schemas and considering the "pragmatic information content" of route instruction texts the same if they produce equivalent actions (2003).

Route instruction generation

Moulin & Kettani's GRAAD software generated a logical, specification of a route from a "Spatial Conceptual Map" and gave them to a virtual pedestrian (1998). This logical formulation was processed by another module to convert it into natural language by removing redundant information, matching logical terms with environment names and matching logical relations with verbs. Stocky's kiosk system

[1] The term "route instruction" avoids confusion with the terms "cardinal direction" (north, west, etc.) and "relative direction" (left, up, etc.), components of route instructions.

EDA	turn to face the green halllway, walk three times forward, turn left, walk forward six times, turn left, walk forward once
EMWC	Follow the grassy hall three segments to the blue-tiled hall. Turn left. Follow the blue-tiled hall six segments, passing the chair, stool and bench, to the intersection containing the hatrack. Turn left. Go one segment forward to the corner. This is Position 5.
KLS	take the green path to the red brick intersection. go left towards the lamp to the very end of the hall. at the chair, take a right. at the blue path intersection, take a left onto the blue path. at the coat rack, take another left onto the plain cement. at the end of this hall at the corner, you are at position 5
KXP	head all the way toward the butterfly hallway, keep going down it until you reach a dead end square area. pos 5 is in the corner to the left as you enter the square block.
TJS	go all the way down the grassy hall, take a left, go all the way down the blue hall until you see a coat rack, take another immediate left.
WLH	from four face the grass carpet and move to the hat rack, turn left and move onto the blue carpet, walk past two chairs and to the lamp, turn left, move into the corner such that the lamp is behind you and to your right you see a gray carpeted alley

Table 1: **Example route instructions by different directors,** all from Position 4 to Position 5 in the environment in Figure 2.

guided visitors to offices with a virtual avatar that combined gestures and natural language route instructions (2002). Porzel, Jansche, & Meyer-Klabunde examine issues of how to linearize a representation of a two- or three-dimensional environment or scene into a one-dimensional string of words (2002).

Modeling route instruction language

The first step of route instruction understanding is to model the language used in the route instructions. A portion of route instructions from a cognitive study were hand-labeled for semantic features (See Table 1 for examples). A *referring phrase* is a phrase that refers to some entity or attribute being described, analyzed on its semantic content instead of its syntactic makeup (Kuipers & Kassirer 1987). By semantically tagging the referring phrases and verbs in a set of route instructions, this analysis characterized the surface meaning of route instruction utterances. From the hand-labeled text, a Probabilistic Context Free Grammar (PCFG) was trained to parse and semantically tag new route instruction texts.

Figure 1 shows the complete framework for the linguistic understanding of verbal route instruction texts. The parser, here from a PCFG, parses the unstructured plain route instruction text to produce a syntactic parse tree annotated with word senses, or meaning in context. This tree can be transformed into an *Attribute Value Matrix* (*AVM*) representation, AVMs are a recursive representation of the surface meaning of an utterance, after taking out incidental features, such as phrase order and word selection. AVMs can represent the information captured in the referring phrase verbatim protocol analysis (Kuipers & Kassirer 1987): verb types and their required and optional arguments.

Finally, the framework integrates the knowledge in individual utterances to extract the meaning of the route instruction text. A powerful representation of the route can be found in the Spatial Semantic Hierarchy, as introduced in the next section. An explanation of the processing of attribute-value matrices into a representation of the route follows this brief introduction to the SSH.

Representing routes in the Spatial Semantic Hierarchy

Route instructions can be represented naturally using the representations of the Spatial Semantic Hierarchy (SSH) (Kuipers 2000). Route instruction texts describe causal and topological structures annotated with metrical and rich view (object and landmark) information. In the SSH, the *Causal* level discretizes continuous control motions into reliable, high-level actions. At the causal level, motions are abstracted to either *turn* or *travel* actions. A turn action changes orientation within a place, while a travel moves the agent from one place to another. A *view* likewise abstracts the sensory image.

The *topological* level of the Spatial Semantic Hierarchy represents the environment as *places*, *paths*, and *regions* and the topological relations of *connectivity*, *path order*, *boundary relations*, and *regional containment*. Likewise, there are topological actions, such as “go to the third place down the path” and “get to the intersection of the brick path and rose-floored path.”

Route instructions can be represented by the SSH causal and topological ontologies, with the actions annotated with metrical and view attributes. Route instructions are expressed in both declarative and procedural language, e.g. both “As you walk down the hall, you will see a lamp,” and “Walk down the hall past the lamp.” Route instructions include both causal actions (“Walk forward three steps.”) and topological actions (“Face along the blue path and follow it to the intersection with the brick hall.”).

Inferring an SSH topological route map

The Attribute Value Matrix is a good representation of the shallow meaning of route instruction text utterances. The remaining step builds the gestalt meaning of the text from the individual utterances. To reason about the semantics (meaning), anaphora (co-reference resolution), and discourse properties (inferring the conversational intent of an utterance) of route instruction texts, I examine how the spatial language is used.

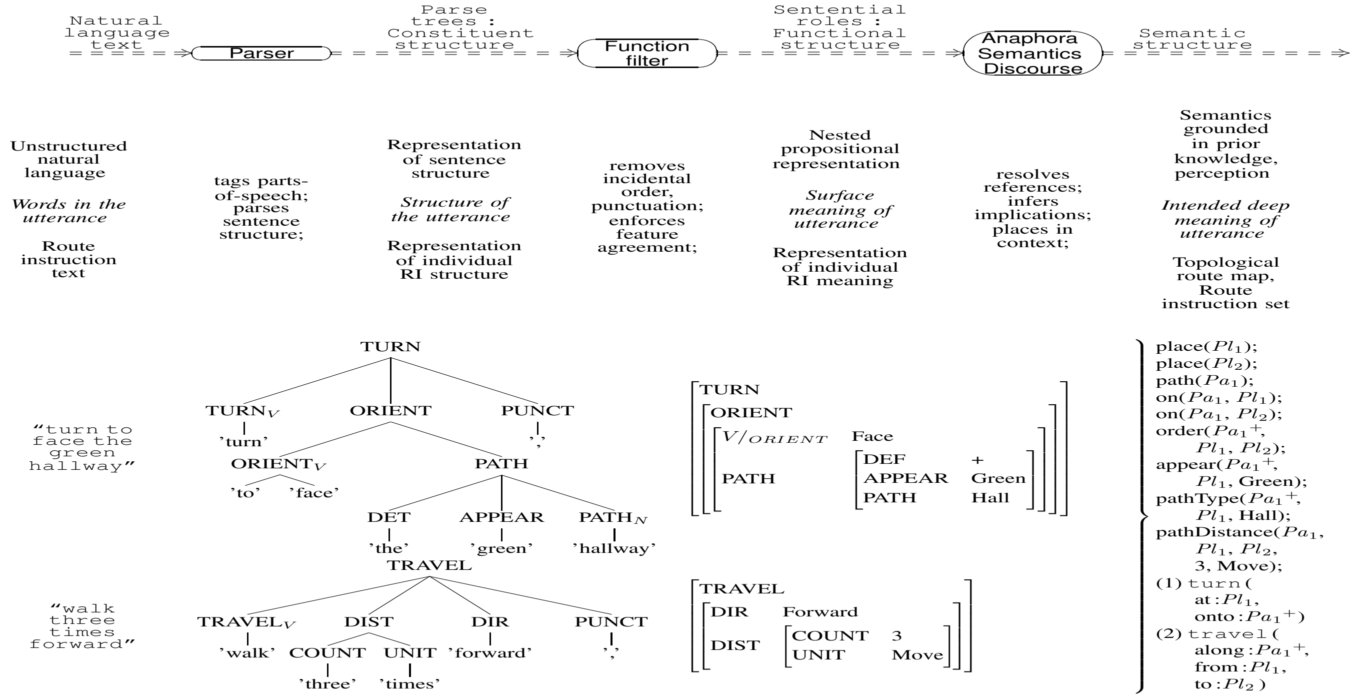
Natural language text
Parser
Parse trees : Constituent structure
Function filter
Sentential roles : Functional structure
Anaphora Semantics Discourse
Semantic structure
Unstructured natural language
Words in the utterance
Route instruction text
tags parts-of-speech; parses sentence structure;
Representation of sentence structure
Structure of the utterance
Representation of individual RI structure
removes incidental order, punctuation; enforces feature agreement;
Nested propositional representation
Surface meaning of utterance
Representation of individual RI meaning
resolves references; infers implications; places in context;
Semantics grounded in prior knowledge, perception
Intended deep meaning of utterance
Topological route map, Route instruction set
"turn to face the green hallway"
TURN
TURN_V
'turn'
ORIENT
ORIENT_V
'to'
'face'
PATH
DET
'the'
APPEAR
'green'
PATH_N
'hallway'
PUNCT
','
"walk three times forward"
TRAVEL
TRAVEL_V
'walk'
DIST
COUNT
'three'
UNIT
'times'
DIR
'forward'
PUNCT
','
TURN
ORIENT
V/ORIENT Face
PATH DEF +
APPEAR Green
PATH Hall
TRAVEL
DIR Forward
DIST COUNT 3
UNIT Move
place(Pl1);
place(Pl2);
path(Pa1);
on(Pa1, Pl1);
on(Pa1, Pl2);
order(Pa1+, Pl1, Pl2);
appear(Pa1+, Pl1, Green);
pathType(Pa1+, Pl1, Hall);
pathDistance(Pa1, Pl1, Pl2, 3, Move);
(1) turn(at:Pl1, onto:Pa1+)
(2) travel(along:Pa1+, from:Pl1, to:Pl2)

Figure 1: **Natural language understanding framework** with an example of understanding a route instruction text.

Both Grice's conversational maxims (1975) and Sperber & Wilson's Relevance Theory (2004) are linguistic theories of discourse – how sentences are strung together to form broader meaning. Each theory assumes that a cooperative speaker conveys meaning by crafting the discourse to clearly and concisely carry across the necessary concepts. In this vein, I propose sets of axioms and heuristics for a follower to infer the topological map. Once the makeup and structure of the topological map is known, the map can be displayed graphically as a user or developer interface. The bottom right corner of Figure 1 shows the propositional topological maps for the short route described by these two sentences. A graphical representation from another route instruction text is shown in Figure 3.

One set of axioms instantiates the entities, or *resources*, required by complex concepts. For instance, "the corner" in these instructions usually refers to the intersection of two paths, each terminating at the corner. Other *resource axioms* include "each path connects at least two places, "each intersection is the meet point of at least two paths", and "a left or right turn action implies changing paths".

Another class of axioms tracks when information is explicitly mentioned and when it can be inferred. These *conversational axioms* include "When two consecutive turn commands specify their location, these are distinct places separated by an unstated travel action." and "When a turn is immediately followed by a travel without a location mentioned, the travel starts where the turn results." These help resolve anaphora issues – is the place or path mentioned in the current sentence new or previously mentioned?

Resolving linguistic anaphora is analogous to resolving *place aliasing*, or perceptually identical places, while exploring. Often, the route instructions do not completely specify the route, leaving spatial ambiguity. For instance, a turn direction may be unspecified, leaving topological ambiguity. Fortunately, the SSH can handle these partial states of knowledge. Moreover, the SSH has been extensively used to resolve the spatial ambiguity present while performing simultaneous localization and mapping. The partial, ambiguous map of the environment derived from language understanding can be handled by exactly the same processes that handle the partial, ambiguous map learned from exploration.

Ideal route instruction models

An empirically tested model reveals which aspects of route instructions lead the follower reliably to the destination. The route can be segmented into *route legs*, where each *route leg* consists of selecting, orienting, executing, and terminating a *topological travel action* that transports the traveler from one place to another.

A route instruction text can likewise be segmented into the utterances describing each route leg. A route consists of a chain of route legs; a route instruction set is a chain of route instructions. Both meet the Markov criterion: after traveling or describing a route leg, the remainder of the route is independent from how the current state was achieved. Once one has described or followed a route to be facing in a direction at a position, the problem is the same as starting a route from the intermediate pose to the destination.

Since the Markov assumption is met, the probability that a set of route instructions will successfully guide the follower

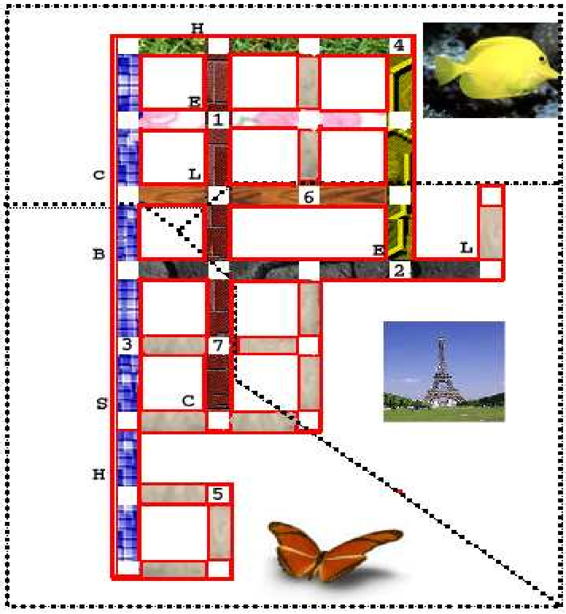

Figure 2: **The "Grid" environment.** Example route instructions drawn (Table 1) from a virtual reality human study in an environment with this layout, 11 instances of 6 objects, 7 floor textures, and three wall-hangings.

along a route to the destination can be estimated by a *Markov Model*. *Markov models* are systems of stochastic transitions between discrete states (Kaelbling, Littman, & Cassandra 1998).With partial knowledge of some of a route leg description, an environment and the follower's cognitive map, perceptive skills, and motive capabilities, such as assuming a generic follower, a *Hidden Markov Model* (*HMM*) can estimate the probability of the route leg being successfully traversed. Each route leg is a link in a Markov chain where the probability of success from that point forward does not depend on the history of what was described and how it was followed.

Use of Markov models serves two purposes: First, the Markov models provide a flexible rigorous framework for describing a variety of followers and directors with different abilities in different conditions. Second, Markov models allow exact or approximate optimal solution that put a bound on performance. These ideal route follower and ideal route director models provide comparison benchmarks for all route directors, like ideal observer models do for perception tasks and the ideal navigator does for spatial localization and navigation tasks.While these ideal models may prove infeasible to fully optimize in some real-world cases, approximate solutions may provide excellent task performance. Even where search for the optimal set of route instructions is intractable, this conceptual framework for evaluating route instructions can guide heuristic search for high quality route instructions.

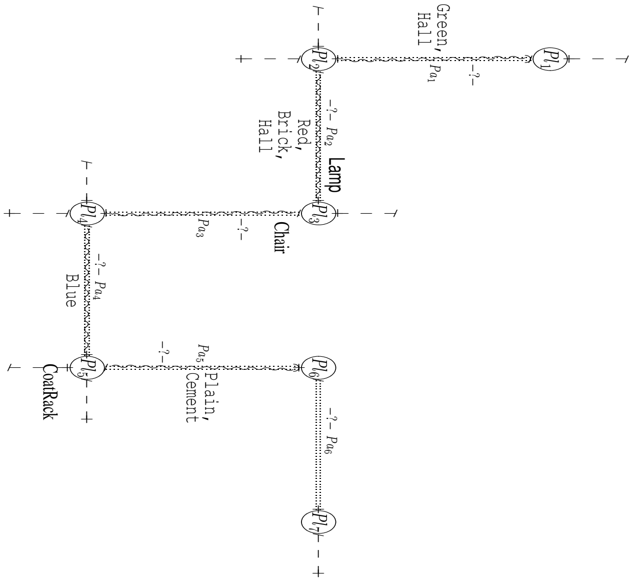

Figure 3: **Topological map** derived from route instructions (KLS, Table 1), Position 4 to Position 5 in "Grid" map.

Markov model notation

I will use the following notation to specify Markov models. As in (Kaelbling, Littman, & Cassandra 1998), a *partially observable Markov decision processes* (*POMDP*) is a tuple $\langle S, A, T, R, \Omega, O\rangle$ where:

S is set of discrete states

A is a set of discrete actions

T is a transition function $T(s, a, s') \rightarrow [0, 1]$ defining the probability of the transitioning from state s to state s' in the next time step, given action a,

R is a reward function $R(s, a) \rightarrow \Re$ mapping each combination of state s and action a into a continuous reward value,

Ω is a set of discrete observations, and

O is an observation function, $O(o, a, s) \rightarrow [0, 1]$, describing the discrete probability distribution over observing o in state s performing action a.

A *hidden Markov model* (*HMM*) consists of a tuple $\langle S, T, \Omega, O\rangle$. Since there are no actions in the HMM, $O$is $O(o, s) \rightarrow [0, 1]$ and $T$is $T(s, s') \rightarrow [0, 1]$.

Ideal route instruction follower

We follow Roy, Pineau, & Thrun in modeling natural language understanding as a Markov process where the hidden state is the speaker's intention. In the Nursebot elderly assistant domain, the robot Pearl plan dialog with a POMDP (Roy, Pineau, & Thrun 2000). The state was the speaker's intended meaning and the actions were physical and speech actions, such as clarifications.

In route instructions, the director's intent is that the follower navigate a certain route. Thus, Markov models capture the trade-offs in crafting short, robust route instructions and handle following an ambiguously described route. The *ideal follower* maximizes the probability of reaching the destination given a route description and any knowledge of the environment or director. However, a navigating agent can be ideal under several different metrics, with differing assumptions.

First, one can optimally guess the meaning of an ambiguous route instruction set. This task can be modeled, like many in computational linguistics, by a Hidden Markov Model. Paring text into a representation of the surface semantics, such as the attribute value matrix, can be represented by a HMM:

S is the set of AVM route leg descriptions;
T is the likelihood of transitioning among AVMs;
Ω is the set of textual utterances; and
O is the likelihood of an AVM given an utterance.

This HMM models the lexicon and route instruction style of a director, that is, how a director uses words and strings together utterance types. For instance, some director's instruction style alternates between turn and travel commands.

Another HMM models the transformation from the surface meaning of sentences to the deep meaning of the route instruction set:

S is the set of (partial) route maps;
T is the likelihood of options to growing the route map;
Ω is the set of AVM route leg descriptions; and
O is the likelihood of a new route map given an AVM.

This HMM models the route knowledge from the utterance's meaning and conversational implicatures. The implicatures are the resource and conversational axioms introduced above. Solving both HMMs maximizes the probability of correctly inferring the map and instruction series the director intended, using models of the syntax, semantics, and pragmatics of the director's route instruction language usage.

These hidden Markov models can be extended with navigation actions to form partially observable Markov decision processes. Assuming a correctly inferred route map and route instruction set, the problem is now Markov localization (Kaelbling, Littman, & Cassandra 1998):

S is the set of inferred places and path segments;
A is a set of inferred travel and turn actions;
T comes from the connectivity of the route map and the likelihood of executing each route instruction;
R is gives a reward for reaching the destination and a penalty for each movement;
Ω is a set of mentioned appearance and layout attributes; and
O is models the likelihood of being on a path segment or at a place given an observation.

When more than one map is possible from an ambiguous route instruction text, the problem is a Markov Simultaneous Localization and Mapping (SLAM). One formulation has the state set S compilation of all states in all possible maps and the transition function T the union of all individual transition functions. Thus, the SLAM problem of distinguishing between possible maps becomes the problem of localizing in space of disconnected, similar regions, one of which is the true map.

The route instruction text does not uniquely specify action sequences, but constrains navigation by providing a plan skeleton, with exploration sub-goals the follower must accomplish. This follower is akin to the examples on route instruction following by Agre & Chapman (1990). Where they explained "plan-as-communication", I explain communication as a partial plan, or, more precisely, a Markov policy.

When the follower interacts with the route director, it can also optimize dialog. The follower decides when to interrupt the route director with a question, trading off the cost of asking a question against the likelihood of missing the destination. To the SLAM formulation above, the action set adds speech acts or dialog moves that clarify, disambiguate, or fill in needed knowledge. Roy, Pineau, & Thrun's Nursebot reasoned about when and how to extend the dialog to increase understanding (2000).

Finally, the follower can decide when to interact and ask for route instructions, balancing the benefit of more likely reaching the destination against the cost of interrupting and querying someone. This adds actions of finding a knowledgeable director and initiating dialog, as well as the state the location of directors. Thus, the most capable agent must optimally chose when to seek help by asking for route instructions, optimally query for route instructions, and optimally navigate using the route instructions.

Discretely adding to the action set A captures progressively more capable agents, which accomplish, in turn, route modeling, route navigation, route dialog management, and route dialog initialization. Discretely adding to the state set S captures reasoning about uncertainty over the most likely route, uncertainty about position on the route, and uncertainty about possible routes. Finally, varying the reward function moves smoothly among follower's devotion to slavishly following the given route (vs. looking for shortcuts), willingness to ask questions of the director, independence in executing the route (vs. finding help), and diligence in reaching the goal with certainty (vs. guessing wildly or giving up short of the goal).

Ideal route instruction director

A partially observable Markov decision process can balance the cost of longer or more complex route instructions against the marginal increase in probability of success. Fist, the director can reason about a known or generic follower correctly following route instructions:

S is the set of places and path segments;
A is a set of spatial and action descriptions;
T comes from the connectivity of the route map and the likelihood of correctly executing each route instruction;
R is gives a reward for the follower reaching the destination and a penalty for each movement and utterance;
Ω **and** O are empty.

The director aims to produce the route instruction set that is the most reliably and easily followed to the destination. If the director expands the transition function T to include the likelihood of understanding the route instructions and the reward function R to cover the cost of understanding, the agent will trade off the costs of text generation and follower cognition against the likelihood of the follower *understanding* and knowingly reaching the destination. By adding observations of the follower navigating or rephrasing the instructions, the director can account for feedback. Finally, action set S can include speech acts such as questions, to actively ensure comprehension or query spatial knowledge. This POMDP can generate the optimal route instruction dialog, adding generating interrogative sentences to the declarative and imperative sentences of a stand-alone set of route instructions.

Conclusions

A computational model of route instructions addresses interesting questions in cognitive psychology, computational linguistics, artificial intelligence, and software engineering. Route instructions are a window into spatial cognition and communication. Discovering how people describe routes informs cognitive science how people communicate and reason about complex spatial sequences. In artificial intelligence, the benefits are grounding verbal symbols from a route instruction text to symbols in the cognitive map and to action in an environment. This project will advance robotic cognitive mapping both by interfacing it with a human communication system and by applying it to new domains.

The engineering benefits include a system that can produce route instructions which are easier to understand and follow. Route instruction giving systems, such as map kiosks, web, phone, and in-car services, can be improved. By recognizing and carefully describing the more difficult segments in route instructions, a system can generate route instruction texts which are more natural, more easy to follow, and more reliable. Another application could check route instruction texts from other sources for completeness, clarity, coherence, and conciseness. By understanding the semantic content of route instructions and the distinctions between good and bad route instructions, a route instruction following system can repair poor route instructions by asking for additional information. For a system following route instruction texts, such as a mobile robot, a smart wheelchair, or a handheld navigation aid, modeling the semantic structure of route instructions will improve text understanding and speech recognition.

One result of this research is a software system that parses meaning from a route instruction text, combines the new knowledge with any previous cognitive map, follows the route instruction set by taking appropriate navigation actions, and finally generates route instruction texts. A system that can parse route instructions to learn about the world and follow a route, while conversely being able to produce route instruction texts which are reliable and easy to follow, has connected linguistic forms with semantic representations grounded in action. Applications for a route instruction understanding system include mobile robots which can both follow and provide verbal route instructions, street route instruction generation systems, and assistive technologies that can guide users through large, complicated spaces.

References

Agre, P. E., and Chapman, D. 1990. What are plans for? *Rob. & Auton. Sys.* 6:17–34.

Coventry, K. R., and Oliver, P., eds. 2002. *Spatial Language: Cognitive and Computational Perspectives*. Boston, MA: Kluwer Academic Publishers.

Frank, A. U. 2003. Pragmatic information content: How to measure the information in a route description. In Duckham, M., ed., *Foundations of Geographic Information Science*. Taylor & Francis. 47–68.

Grice, H. P. 1975. Logic and conversation. In Cole, P., and Morgan, J. L., eds., *Speech Acts*, volume 3 of *Syntax and Semantics*. New York: Academic Press. 43–58.

Kaelbling, L. P.; Littman, M. L.; and Cassandra, A. R. 1998. Planning and acting in partially observable stochastic domains. *AI* 101:99–134.

Kuipers, B., and Kassirer, J. 1987. Knowledge acquisition by analysis of verbatim protocols. In Kidd, A., ed., *Knowledge Acquisition for Expert Systems*. New York: Plenum.

Kuipers, B. J. 2000. The Spatial Semantic Hierarchy. *AI* 119:191–233.

Moulin, B., and Kettani, D. 1998. Combining a logical and an analogical framework for route generation and description. *Annals of Mathematics and Artificial Intelligence* 24(1-4):155–179.

Müller, R.; Röfer, T.; Lankenau, A.; Musto, A.; Stein, K.; and Eisenkolb, A. 2000. Coarse qualitative descriptions in robot navigation. In Freksa, C.; Brauer, W.; Habel, C.; and Wender, K. F., eds., *Spatial Cognition*, volume 1849 of *LNCS*, 265–276. Springer.

Perzanowski, D.; Schultz, A. C.; Adams, W.; Marsh, E.; and Bugajska, M. 2001. Building a multimodal human-robot interface. *IEEE Intelligent Sys.* 16–21.

Porzel, R.; Jansche, M.; and Meyer-Klabunde, R. 2002. Generating spatial descriptions from a cognitive point of view. In Coventry and Oliver (2002). 185–208.

Riesbeck, C. 1980. "You can't miss it!": Judging the clarity of directions. *Cog. Sci.* 4:285–303.

Roy, N.; Pineau, J.; and Thrun, S. 2000. Spoken dialog management for robots. In *Proc. of 38th Ann. Meeting of the ACL(ACL-00)*. Hong Kong, China: Morgan Kaufmann.

Simmons, R., et al. 2003. GRACE: An autonomous robot for the AAAI Robot Challenge. *AI Magazine* 24(2):51–72.

Sperber, D., and Wilson, D. 2004. Relevance Theory. In Horn, L. R., and Ward, G., eds., *The Handbook of Pragmatics*. Oxford: Blackwell. 607–632.

Stocky, T. A. 2002. Conveying routes: Multimodal generation and spatial intelligence in embodied conversational agents. Master's thesis, Mass. Inst. of Technology, Cambridge, MA.

Webber, B.; Badler, N.; Di Eugenio, B.; Geib, C.; Levison, L.; and Moore, M. 1995. Instructions, intentions and expectations. *AI* 73(1–2):253–269. Spec. Issue on "Compl. Res. on Interaction and Agency, Pt. 2".