

When is a union really an intersection? Problems interpreting reference to locations in a dialogue system

Myroslava O. Dzikovska and Donna K. Byron

Department of Computer Science
University of Rochester
Rochester NY 14627, U.S.A.
myros/dbyron@cs.rochester.edu

Abstract

This paper describes issues that arose in our implementation of an interpreter for locative expressions in a spoken dialogue system. The expressions involve complex adverbial modification, include imprecise and inaccurate content, and reflect conventional practices specific to the way English speakers refer to roads and highways. Our system utilizes semantic features in the parser to perform disambiguation, and domain-specific reasoning to resolve variables in the logical form to the correct referent. The functionality implemented to date was evaluated against a small corpus of naturally-produced expressions, of which 46% were correctly interpreted.

1. Introduction

This paper describes issues that arose in our implementation of an interpreter for locative expressions in the TRIPS-911 spoken dialogue system (Ferguson and Allen, 1998). In this system, the user is faced with a city map (Figure 1) and he must act as the dispatcher in an emergency-response center. Unlike our previous domain, in which the user could only refer to named map locations such as cities, the detailed nature of the new map necessitates that locations be described rather than named. Map objects described during a typical session include intersections (e.g. “where Main and Oak cross”), road segments (e.g. “all of Monroe between Main and Oak”), and regions of town (e.g. “in the north of the city”). Even objects with names, such as hospitals, may be described rather than named by users that are unfamiliar with the map.

In building an interpretation component for these expressions, our goal is to support natural language use in which many different expressions can describe the same object, and in which descriptive terms can be embedded to an arbitrary degree. To interpret these expressions requires first generating a logical form that accurately represents the semantics of the sentence, then using that logical form to constrain the search for the correct referent(s) of the expression.

Many reference resolution studies start from the logical forms, without considering the problem of relating them to natural language utterances. In building a practical dialogue system, these processes need to work together in order to quickly and accurately interpret the descriptions. This paper describes the entire process from natural language utterance to logical form to reference resolution. We first discuss some



Figure 1: Map used in the TRIPS-911 system

of the problems that exist in trying to interpret spoken natural language descriptions of locations; after discussing the relevant background, we then describe the interpretation process for locatives in our system, and evaluate the portion of the design that has been implemented to date.

2. Motivation

In preparing to develop a system to converse in this new domain, we collected a set of human-human problem solving dialogues involving emergency response tasks such as plowing roads, dispatching medical personnel, and repairing downed electrical lines (Stent, 2000). One of the most interesting aspects of these dialogues is the large variety of expressions that were used to refer to map objects. We were impressed

1) “the intersection of three eighty three and two fifty two A just below the airport”
2) “where Genesee, Brooks Road and three eighty three connect”
3) “at route thirty one at three ninety”
4) “the corner of Main street and East avenue downtown”
5) “at route two fifty two and the river”

Table 1: A variety of referring expression forms indicating junctions

1) “directly east of the inner loop a little bit”
2) “quite in the north”
3) “over near Gates”
4) “at three eighty three just past two fifty two A”
5) “the bridge near Gates where four ninety crosses the river there”

Table 2: Expressions involving imprecise descriptions and multiple constraints

by the richness of expressions we found in this corpus compared to the previous domain, which used an extremely simple (and fictional) map of a small island. A cursory examination of the corpus showed that complex locative expressions occur very frequently in this domain, and that special reasoning needed to be developed in the system to cover them. This section presents a few selected example phrases from our corpus to show the difficulty of developing automated methods to interpret these expressions.

References to streets, highways and junctions where streets cross can take many forms, some of which are shown in Table 1. These sentences demonstrate many of the problems we must tackle in converting all these different surface forms into the same logical form. From these examples, we can see that junctions can be indicated explicitly with nouns like “intersection” or propositions like “cross” and “connect”, and implicitly, using a conjunction in the context requiring a location, such as the argument of the preposition “at”. Junctions can involve just two or sometimes more than two streets, or even a street and another ribbon-shaped object such as a river.

Streets are often referred to by a proper name, such as “Main Street”, so our intuition would be that streets can be processed simply as proper names, but this is too simplistic for a real application. Towns tend to include several streets of the same name, for example Genesee Avenue, Genesee Court, and Genesee Lane might all exist on the same map. When the speaker of sentence 2 says simply “Genesee”, we have to determine which street he means, either by applying constraints from the remainder of the description (eg. which street named Genesee intersects with Brooks) or by applying a heuristic (prefer main roads over side streets). Also, in sentence 2, Brooks Avenue is incor-

rectly referred to as Brooks Road. We would like the system to be able to resolve this reference nonetheless.

Reference to highways in this domain have an additional set of problems because highways in the United States are typically given multi-digit numerical indicators. This introduces additional ambiguity, because depending on the context the numeral 104 (pronounced ‘One oh four’) can be interpreted as the highway 104, a time-point 1:04, maybe even an office number or some other kind of entity. Selecting the proper interpretation presents a challenge for a spoken dialog system.

Expressions that describe the location of an object on the map can be arbitrarily complex and often use imprecise operators. Converting these operators into well-defined functions in the interpretation process is not straightforward. Table 2 shows several such phrases from our corpus using the operators “east”, “north”, “near” and “past”. These operators can be further modified by vague adverbials such as “just”, “slightly” or “directly”, which further limit the space from which the referents can be selected.

In the following sections we present some of the approaches we used to deal with the kind of descriptions presented above.

3. Background and Related Work

Much of the previous work in linguistics and psycholinguistics on human production and comprehension of location descriptions have focused on generating proper scene descriptions (cf. (Retz-Schmidt, 1988)). We found little of that work to be helpful in implementing an actual interpretation mechanism. Mainly this is because our system does not resolve spatially-located reference (objects located in a 3-D space and requiring the calculation of reference points,

relative scale, etc.). Our requirements are simply to process locative expressions in a 2-D space from a fixed perspective. Also, many studies start from a disambiguated logical form rather than from natural language utterances, and do not relate their semantic theories with the needs of a parser.

Based on the study of spatial prepositions in different languages, Talmy (1983) identifies a set of possible spatial idealization schemas and properties that we associate with objects, such as idealization to a point, line or strip. He also enumerates the primary relations that can connect the objects in a scene and the possible restrictions on their arguments. Herskovits (1986) argues that simple relations are insufficient to explain the way locatives are used. She uses a similar set of schemas and properties and develops a formal geometric scene representation. In her theory, every preposition has an “ideal meaning” which is then transformed into actual meaning with the aid of pragmatic principles such as relevance, salience, topicality and tolerance. In a sense, our approach is similar, because the parser generates underspecified predicates in the surface form that can be seen as idealized meanings, and then the reference agent computes their actual meaning based on pragmatic considerations. However, her principles as described in the book are rather too vague to treat computationally.

Creary et al. (1989) present a logic of location predicates suitable to use in reference resolution. This work represents locative constraints as expressions over regions composed using intersection and inclusion operations. Their representation is computationally efficient and deals with scope ambiguity, permutability and omissibility of locatives. We use a similar idea in our reference resolution algorithm. However, Creary assumes a fully disambiguated logical form, and therefore does not specify how to convert utterances into logical forms. In our case, the parser and reference resolution need to do extra work disambiguating the surface predicates.

4. The interpretation process in the TRIPS system

The TRIPS parser uses a chart-based best-first parser loosely based on the HPSG grammar formalism and described in (Allen, 1995). The parser receives a string of words from speech recognition or the keyboard and then obtains a syntactic analysis and a surface logical form. The final logical form includes tokens representing the binding for each referring expression. The reference agent then returns the object it believes the logical form refers to, along with the score indicating its confidence in the resolution result. In

this section, we first describe the process by which the parser converts each referring expression into a logical form. Then we discuss how the reference resolution agent (RA) uses this logical form to resolve the referring expression to the correct entities.

4.1. Generating a Logical Form

Since the TRIPS parser is connected to an interactive system, speed and accuracy is of the essence. For the sake of efficiency, we would like to have as much disambiguation as possible to be done on early stages of interpretation, preferably during parsing, to eliminate (slow) reasoning about implausible analyses. On the other hand, we would like to keep the system portable to other domains, and therefore need to have a grammar and lexicon that are mostly domain-independent. As a way to balance these criteria, our system is heavily dependent on semantic selectional restrictions¹ that work to keep the parsing complexity down and to help disambiguate syntactic structure. An alternative to this solution would be implementing a statistical method to select among possible interpretations. However, statistical methods require large amounts of text for training, and this is not available in our domain. Moreover, the constructs we encountered in our corpus are not often found in more formal sources such as Wall Street Journal, precluding their use for purposes of training at this time.

Therefore, in designing the lexical semantic representation we have to worry not only about selecting a semantic representation that would accurately express the meaning of a word, but also about formulating the selectional restrictions that are useful for disambiguation. From the point of view of the system development, we found that separating those two issues to some extent helps to make lexicon maintenance and development easier. In order to do that, each word in the lexicon is characterized with a predicate, which corresponds to the (deep) meaning of the word and can be mapped to the corresponding entity in the domain knowledge representation, and a set of semantic features that express some basic lexical semantic properties of the word meaning that are used in formulating selectional restrictions and disambiguation.

Our initial feature set included most of the EuroWordNet top hierarchy features (Vossen, 1997), with the value sets somewhat modified to suit our needs. However, we discovered that these features were inadequate to provide reasonable selectional re-

¹While selectional restrictions have a variety of problems that make their use in a general case impossible, we believe that they are a useful mechanism to control parser complexity in a particular domain.

at preposition

LF AT-LOC ;; the meaning predicate to be used in the logical form

SUBCATSEM (spatial-abstraction point) ;; the semantic restriction on the subcategorized NP

ARGSEM (spatial-abstraction point) ;; the semantic restriction on the object modified by the PP

LF AT-TIME

SUBCATSEM (function time-object)

ARGSEM (Aspect Bounded)

bridge noun

LF BRIDGE

SEM ;; the semantic features associated with the word

(spatial-abstraction (OR point line)) (origin artifact)

(form geographical-object) (function location)

Brooks name

LF ROAD ;; for names LF carries the type of object to look for

NAME BROOKS-AVENUE ;; the constant corresponding to the object

SEM

(spatial-abstraction (OR point line strip)) (origin artifact)

(form geographical-object) (function location)

Figure 2: Parts of lexical semantic representation used to disambiguate “the bridge at Brooks”.

restrictions on locative predicates, so we augmented the set with *spatial-abstraction* and *form* features inspired by the work of Talmy and Herskovits. The selection of features is based on the idea that people abstract the actual shapes of objects to a (small) set of abstract geometric shapes, and these abstractions restrict what locatives can be used in connection with the scene.

Consider the sentence “Go to the bridge at Brooks”. Among other things, in interpreting that sentence we need to decide that *at* in this context has locative meaning and that the prepositional phrase “at Brooks” modifies the noun phrase “the bridge.”²

Figure 2 contains partial definitions for the words *bridge*, *at* and *Elmwood*.

The definition of the name Brooks states that it can be visualized as a point (in our system, all geographical objects can). This helps the parser to distinguish the AT-LOC sense of *at* from other possibilities, e.g. AT-TIME³. Moreover, the definition of AT specifies

²It is also possible for the “at” to modify the verb “go”. In our domain this rarely happens and the lexical entry for “go” is set up so that this interpretation is excluded for the reasons similar to those discussed below

³our representation is set up so that the features corresponding to physical objects, such as (*spatial-abstraction point*) and features corresponding to abstract entities such as (*function time-object*) are mutually exclusive, which allows us to make this inference on the basis of information highlighted in the picture

that it can only modify the objects that themselves can be visualized as points, and the definition of the *bridge* satisfies this condition. Thus, the parser sends the following objects to the RA to resolve the locative expression “the bridge at Brooks”:

```
x: (AND(TYPE x ROAD)
      (NAMEOF x BROOKS-AVENUE))
y: (AND(TYPE y BRIDGE) (AT-LOC y x))
```

There is still a problem with defining selectional restrictions in terms of features in our system. Obviously, the granularity for spatial reference varies with the task. Depending on the scale, any physical object can be assigned almost any *spatial-abstraction* value. Consider words like *intersection* or *truck*. In a different domain, it is possible to imagine them visualized as areas in space, with something located ACROSS or OVER them. However, at the scale accepted in our domain, intersections and trucks are points, and such expressions do not often occur. If the selectional restrictions are loose enough to allow those other phrasings in our domain, the ambiguity of spatial expressions increases noticeably, requiring additional reasoning to disambiguate the obtained logical forms. Therefore, we selected a specific scale suitable for the domain, even though this excludes some of the interpretations allowable in other domains. Not surprisingly, this rigid restriction sometimes results in excluding expressions that should be acceptable in our task. The

solution could be to introduce softer selectional preferences, which is planned as a part of our future work in the system.

4.1.1. Pragmatic considerations

One can note that the feature-based representation is not suitable for fully representing spatial properties of an object (cf. (Jackendoff, 1983)). In fact, it is not the intended use of features in our system. A small set of features is obviously not sufficient to express all possible distinctions needed to obtain a fully disambiguated logical form. However, in the process of developing the TRIPS system we found that using a general knowledge representation that would allow us to obtain a completely disambiguated logical form before giving it to the reference resolution agent was costly computationally, and was making our lexicon too difficult to maintain. We use the semantic features as a representation that provides the information to cut down the number of possible sentence interpretations, and some basic properties often used by reasoning components, but that does not attempt to encode all the distinctions needed for full disambiguation.

For example, the preposition *from* is often ambiguous. For the phrase “an ambulance from Pittsford”, the features of the word *Pittsford* are sufficient to eliminate the FROM-TIME sense of *from*. On the other hand, this phrase can mean an ambulance currently located in Pittsford, or an ambulance based in Pittsford, and the distinction is determined mostly by context and not by the form of the utterance. Therefore, it cannot be expressed with the kind of selectional restrictions used by the parser. For these cases, the parser outputs the predicate FROM, and the RA must decide on whether the correct interpretation is ORIGIN or AT-LOC, the domain predicates encoded in the general knowledge base.

Another case that requires special reasoning to be implemented in resolution is a practice of referring to highways by their numbers mentioned earlier. Selectional restrictions may be able to eliminate some of the ambiguity, for example, “the bridge on 104” definitely refers to a location. However, this is often not possible, for example, when “one oh four” is uttered as a short answer to the previous question. In this case, our parser will not select a type for numeric expressions but will leave that disambiguation up to the RA, which can use the context to determine the correct interpretation.

Yet another complication resulted from our assumption that street designations could be taken as proper names⁴. As described in the motivation sec-

tion, users can often use abbreviated names that are ambiguous between a number of streets in the same town, or make mistakes in street names. Therefore, we treat street designations more like definite descriptions than proper names. Lexical entries for streets contain the full name, for example BROOKS-AVENUE. If the speaker uses the entire correct name, the parser interprets it as a name. If the user says “Brooks Road” but only “Brooks Avenue” is in the lexicon, the parser will generate a representation (AND (TYPE x ROAD)(ASSOC-WITH x BROOKS-AVENUE)) (where the predicate ASSOC-WITH denotes some association between objects) and let the RA decide how the objects are associated. The RA includes the heuristic that in our domain if two locations are associated, then they share at least some space, and this allows it to make the inference that the road in question is indeed Brooks Avenue. A more difficult case is when objects of different types share a name, (e.g. a street and a body of water named “Ontario”) selectional restrictions may filter out some possibilities. But if more than one interpretation is left and the tie cannot be resolved by the weights associated with grammar rules, the system may simply arrive at an incorrect interpretation. Currently the problem is partially solved by trying the two most likely parses and using interpretation confidence scores to determine which object is a better referent. We plan to modify the system in the future so that the parser outputs a more general type for these cases, such as (TYPE x LOCATION), allowing the RA to reason out the correct referent.

4.2. Interpreting the logical form

Once a logical form representing the referring expression is selected, it is passed to the RA to be resolved. The logical form includes the semantic type of the variable and the description constraints needed to select the referent from among the objects of that type. The RA has access to a geographical database containing all objects on the map, and each referring expression is resolved to one or more map objects. Simple bindings can be resolved by table lookups, e.g. (AND (TYPE w CITY) (NAME-OF w GATES)) is resolved by finding an object of type CITY and named GATES in the map database.

The RA contains domain-independent reasoning to correctly interpret different referring expression forms (eg. definite versus indefinite phrases), and has access to a domain-specific reasoner. This domain-specific reasoner is employed to determine which map objects satisfy the predicates given in the logical form, a pronames is uncontroversial.

⁴Creary *et al* claim that the analysis of streets as proper

cess that varies based on the task to which the system is currently being applied. For the “ambulance from Pittsford” example presented above, it will try to locate both the ambulances currently in Pittsford and those originating from Pittsford, and if more than one entity is returned, will sort the results according to a heuristic taking into account salience and contextual factors.

Additionally, the domain-specific reasoner is responsible for translating the more domain-independent representation produced by the parser into the representation that matches the way objects are stored in a geographical database. For example, in this domain the predicate MIDDLE (e.g. “in the middle of Elmira”) actually means (NEAR (CENTER ELMIRA)), not the epicenter of Elmira, and the precision with which objects can be considered in the middle of another object varies with the domain. The domain-specific reasoner available to the RA translates MIDDLE into its domain-specific use. Other operators may need to be coerced based on the internal representation used in the map. The expression “the end of highway five ninety” produces a logical form (END x highway-590). In our system, the database contains information about the segments that together constitute a road, but there are no facts in the database in the form (END x y). Therefore the domain reasoner must know that to find the END of a road, you must find a point on the highway that is the end of one road segment and that is not the beginning of another segment.

Many locative predicates are interpreted as regions. There are standard techniques to judge whether a point belongs to a region (cf. Gapp (1994)). See Figures 3 and 4 for a simple example of region definitions that represent several variants of the predicate NORTH-OF defined in the geographical database, depending on the shape of the reference object. As the first drawing shows, the predicate NORTH-OF is not interpreted strictly as describing a line running due north from the reference object. The region is relaxed to within plus or minus a few degrees, and objects in this region are considered to satisfy NORTH-OF. The predicate (NORTH-OF x POLICE-STATION1) will be interpreted as the set of objects that fall within this region using POLICE-STATION1 as the reference object. Some modifiers impact the shape of this region, for example (DIRECTLY NORTH) narrows the width of the region, and WAY requires matching objects to be in the upper portion of the region. When the reference object is not a point, the NORTH-OF defines a region along the northern boundary of the object.

Vague modifiers such as “just”, “slightly” or

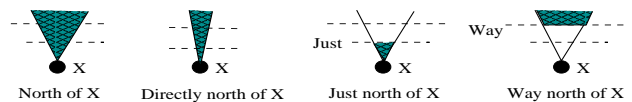


Figure 3: Definition of NORTH if X is a point

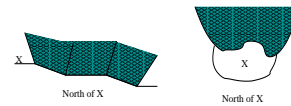


Figure 4: Definition of NORTH if X is a line or a region

“quite” are applied to objects in the list when assigning the RA’s confidence in the object as the correct referent. The confidence scores of objects that satisfy the description ((SLIGHTLY NORTH-OF) x GATES) would be assigned so that items close to Gates have a higher confidence, while the objects satisfying ((WAY NORTH-OF) x GATES) would be sorted so that objects further away have the highest confidence.

4.2.1. Combining the constraints

Earlier we presented some examples where locative descriptions can contain an arbitrary number of constraints. The RA processes these constraints in bottom-up order based on the embedding, which typically corresponds to right-to-left order in the surface expression. Table 3 contains a sample resolution for “The bridge near Gates at four ninety and the river.”

A wide variety of expressions can be used to talk about intersections (see Table 1 above). Many of these constructions can be fully interpreted by the parser, in which case it sends the RA a fully specified intersection description of the form (AND (TYPE z JUNCTION) (JUNCTION z y)), where y corresponds to a set of objects forming the junction. In other cases, additional reasoning is required. For example, in our current implementation, it is difficult for the parser to decide whether the word “and” creates the set with a union of two streets, like “Go down Monroe and Elmwood”, or refers to their common point, as in “it is located at Monroe and Elmwood”. Therefore, the fragment “STREET1 and STREET2” is initially interpreted as a type UNION, corresponding to a set, but the RA knows that in this domain, a union of two ribbonal objects corresponds sometimes to the point at their intersection. As long as the parser can identify the entity as some type of LOCATION rather than a ROUTE or true set (as in the last line of Table 3), a junction object will result. In cases where the junction is described by a proposition (e.g. “where Genesee

Request	Action
Resolve x: (TYPE x RIVER)	Returns objects of type RIVER
Resolve y: (AND (TYPE y ROAD) (NAME-OF y 490))	Returns one item, the road named 490
Resolve z: (UNION z (x y))	The RA makes a set containing street1 and street2
Resolve w: (AND (TYPE w CITY) (NAME-OF w GATES))	Returns one object, the city Gates
Resolve v: (AND (TYPE v BRIDGE) (NEAR v w) (AT-LOC v z))	Returns all bridges matching this form. Now that z is described as the location of v, we know it is a JUNCTION. At this outermost level is an error signaled if the result is still a set.

Table 3: Sample Resolution Process for the phrase:“The bridge near Gates at four ninety and the river”.

and Elmwood connect”) the agent would need to have a set of rules that locate the intersection based on the set of objects mentioned in the description.

5. Evaluation

For our evaluation, we examined 3 dialogues from the Monroe corpus(Stent, 2000), and isolated out all phrases that contained location descriptions. The dialogues included 559 utterances and the resulting list of location descriptions contained 196 phrases (including some duplication). We tested the portion of our interpretation process implemented to date on these expressions. In this evaluation, the expressions were treated (and evaluated) as stand-alone expressions rather than as connected text. Expressions that were anaphoric in the original dialog were judged as correct if the RA bound the variables to map objects matching the descriptive content of the expression. The current implementation includes parser coverage for numeric highway names and underspecified street names (such as “Genessee” for Genessee Avenue). The domain-specific reasoner for reference resolution at this time resolves logical forms for intersections, the end of roads and highways, map directions NORTH/ABOVE, SOUTH/BELOW/UNDER, EAST, and WEST, and FROM.

Table 4 contains the results of this evaluation. The parser found a syntactic analysis and produced a logical form suitable for further interpretation in about 2/3 of the cases. Out of these, we found the correct referents for 92 expressions. Expressions that could not be resolved typically involved unimplemented predicates such as NEAR and various uses of the ASSOC-WITH predicate. Not surprisingly, we are doing better on short phrases that contain fewer modifiers. At the same time, a reference resolution algorithm in the old TRIPS system that could only look up the named locations that are defined in the lexicon could resolve only 47 of those expressions, so adding the special processing for locatives is helpful in our system.

	Number of phrases	Words per phrase
Total	196	5.90
Correctly parsed	131	5.07
Acceptable partial parse	6	10.17
Incorrectly disambiguated	13	8.31
No parse found	46	7.02
Found correct referents	92	4.59

Table 4: Evaluation Results

6. Future Directions

Currently, we have only implemented a small number of predicates. Adding more functionality to the RA, including implementing confidence sorting based on the linguistic hedges is will be accomplished in the near future.

An important problem to be solved is that many locative descriptions produced by people are redundant, and some constraints can be dropped without impeding understanding. Such locative expressions are intended to give a human addressee a visual clue to locate the referent on the map, but are not strictly necessary for the system to understand the reference. These expressions should be applied as soft constraints on the resolution process; handling them properly requires discourse-level processing. For example, there is only one Beahan Road on our map, but the descriptions like “Beahan Road just below the airport” show up quite often in the corpus. Since the name “Beahan Road” can be resolved uniquely, the system could just stop there without checking the rest of the description. A better strategy would be to try to make sure that the object that was found satisfies the rest of the constraints. If this is not the case, this may indicate that the user is confused or simply that the system’s idea of ‘just below the airport’ differs from the user’s. In that case, a clarification a move could be generated, something like “Is this the road you mean?” (the road blinks). We could not implement this strategy due to deficiencies in the system dialogue manager,

but we are currently in the process of implementing a new system architecture (Allen et al., 2000) that would make such processing possible.

Another problem results from the fact that objects on the map being used in the current task cause nearby objects to become salient, and our current model of context management does not account for this. For example, in some of the tasks presented to the users in the Monroe dialogues, they had to evacuate the people from the North-West corner of the city. As soon as this was established, often just by locating an object in that area, people would start using expressions like “go to the end of 390”. This expression is ambiguous, because the road has two ends, and neither has been previously mentioned. However, in those cases it appears that the competitors outside of the region at which the attention is centered are not salient and therefore these references are understood as unambiguous. Therefore, a reference resolution algorithm in this domain should not only take into account recently mentioned objects, but also incorporate information from other sources, for example, the regions where the user tends to look or regions adjacent to the objects on the map that have been recently brought to the user’s attention in the process of creating a plan. Extending the definition of salience in this manner would help us to resolve the anaphoric expressions similar to those mentioned above, and is planned as an important part of our future work.

7. References

- J. Allen, D. Byron, M. Dzikovska, G. Ferguson, L. Galescu, and A. Stent. 2000. An architecture for a generic dialogue shell. *To appear in the Natural Language Engineering Journal special issue on Best Practices in Spoken Language Dialogue Systems Engineering*.
- J. Allen. 1995. *Natural Language Understanding, 2nd edition*. Benjamin/Cummings, Menlo Park.
- L. Creary, J. Gawron, and J. Nerbonne. 1989. Reference to locations. In *Proceedings of ACL '89*, pages 42–50.
- G. Ferguson and J. Allen. 1998. Trips: An intelligent integrated problem-solving assistant. In *Proceedings of AAAI '98*.
- K. P. Gapp. 1994. Basic meanings of spatial relations: Computation and evaluation in 3d space. In *Proceedings of AAAI '94*, pages 1393–1398.
- A. Herskovits. 1986. *Language and Spatial Cognition: an Interdisciplinary Study of the Prepositions in English*.
- Ray Jackendoff. 1983. *Semantics and Cognition*. Number 8 in Current Studies in Linguistics. MIT Press, Cambridge, MA.
- Gudula Retz-Schmidt. 1988. Various views on spatial prepositions. *AI Magazine, Summer 1988*, pages 95–105.
- Amanda Stent. 2000. The monroe corpus. Technical Report 728, University of Rochester Computer Science Department.
- L. Talmy. 1983. How language structures space. In *Spatial Orientation: Theory, Research, and Application*.
- P. Vossen. 1997. Eurowordnet: a multilingual database for information retrieval. In *Proceedings of the Delos workshop on Cross-language Information Retrieval*, March.