Semantics and Pragmatics of Spatial Reference

Dave Golland

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2015-23
http://www.eecs.berkeley.edu/Pubs/TechRpts/2015/EECS-2015-23.html

May 1, 2015

Semantics and Pragmatics of Spatial Reference

by

David Simon Golland

B.S. (Cornell University) 2008

A dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Computer Science

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:

Professor Dan Klein, Chair
Professor Trevor Darrell
Professor Thomas L. Griffiths

Fall 2013

The dissertation of David Simon Golland is approved.

Chair Date

Date

Date

University of California, Berkeley
Fall 2013

Semantics and Pragmatics of Spatial Reference

David Simon Golland

Abstract

Semantics and Pragmatics of Spatial Reference

by

David Simon Golland

Doctor of Philosophy in Computer Science

University of California, Berkeley

Professor Dan Klein, Chair

In order for a robot to collaborate with a human to achieve a goal, the robot should be able to communicate by interpreting and generating natural language utterances. Past work has represented the meaning of natural language in terms of a given database of facts or a simple simulated environment, which simplifies the interpretation or generation process. However, the real world contains a richness and complexity missing from simple virtual environments. Databases present a distilled set of logical relations, whereas the relations in the physical world are more vague and challenging to discern.

In this thesis, we relax these restrictions by presenting models that interpret and generate utterances in a physical environment. We present a compositional model for interpreting utterances that uses a learned model of lexical semantics to ground linguistic terms into the physical world. We also present a model for generating utterances that are both informative and unambiguous. To address the challenges of communicating about a physical world, our system perceives the environment with computer vision in order to recognize objects and determine relations. We focus on the domain of spatial relations and present a novel probabilistic model that is capable of discerning the spatial relations that are present in a physical scene. We establish the functionality of our system by deploying it on a robotic platform that interacts with a human to manipulate objects in a physical environment.

_____

Professor Dan Klein
Dissertation Committee Chair

Contents

List of Figures

List of Tables

Acknowledgements

Some say that completing a PhD is a long and grueling process; and although it had its ups and downs, looking back, the time seems to have gone by quickly and enjoyably. I owe that to the people I met and who have helped me along the journey.

Thank you, Dan Klein, for being an extremely effective advisor. You always have a vision for how things can be improved, whether it is the broad direction of a research agenda or the specific phrasing of a particular sentence, you seem to have an accurate vision for the next steps to make things better. I have been spoiled by your ability to quickly context switch. Whenever one of your students asks you a question, you seem to instantly jump to the proper mental location to generate an appropriate response; I have since learned the hard way that not everyone possesses this superpower. Lastly, thank you for training such impressive students, whom I have had the pleasure to work with.

Thank you, Percy Liang, for mentoring me in the beginning of my academic career. You helped build my confidence and showed me how to think technically about new research problems. While working with you, I caught a glimpse of the field through your eyes. I admire the clarity with which you see concepts as building blocks waiting to be combined to construct the next big thing.

Thank you, Sergio Guadarrama, for always being willing to discuss my ideas and help me find the holes in them. I appreciate having worked with you and Lorenzo Riano, Daniel Gohring, Yanquing Jia, Trevor Darrell, and Pieter Abbeel. It was fantastic to see the preposition model put onto the PR2 robot, the work on which most of the robotics chapter was based. It could not have been done without yours and the others' hard work.

Thank you especially to Trevor, for guiding how my research can be integrated in multi-modal projects. Thank you, Tom Griffiths, for thoughtful comments on drafts of this thesis.

Thank you, Taylor Berg-Kirkpatrick and David Hall, for always helping me fix my models when I was totally lost about where the bugs would be. I am particularly appreciative of Taylor's unbridled enthusiasm about research in general, as well as my specific research — it is tremendously encouraging. Greg Durrett's boundary-less humor, Jono Kummerfeld's thoughtful insights, Adam Pauls' humorous insights, and David Burkett's insightful humor always made working in the bay an interesting and fun experience.

Thank you, John DeNero and Jakob Uszkoreit, for an amazing internship, where I learned a lot and had lots of fun.

Thank you to my "batch mates," Jon Barron, Aditi Muralidharan, Ahn Pham, and Jeremy Maitin-Shepard. We bonded over our "crippling fear / uncontrollable sobbing," during prelim study, which, to be completely honest, I enjoyed immensely. Thanks to Nick Hay for reading groups, Thai food, and thought provoking conversations — all of which were thoroughly enjoyable and kept me learning outside of my area of study. Thank you Mohit Bansal, who was there every step of the way, always reassuring me when I needed it.

Thank you, my Ashby housemates: Fabian Wauthier, Garvesh Raskutti, Lester Mackey,

and Percy Liang. We had lots of chocolate and had lots of interesting conversations about research, philosophy, life, and whole wheat.

Chapter 1

Background

1.1 Scope

The ultimate goal of Natural Language Processing is to engineer a computer system that can understand and respond to commands and queries issued in natural language. The classical examples of natural language systems study the way language relates to itself (Weizenbaum, 1966) or to a virtual world (Winograd, 1972). In this thesis, we present a system that explores how language is used to interact with the physical world. The system can both interpret and generate references to objects in a physical scene, primarily focusing on reference to objects via spatial relations.

Why spatial relations? The focus on spatial relations is motivated by their importance, convenience, and complexity. Spatial relations are universal because they are present in every physical context, and are therefore worthy of study in their own right. It has been argued that spatial relations provide a solid pre-linguistic ground upon which people build the meanings of more complex linguistic concepts (Lakoff and Johnson, 2008). The motivating belief adopted in this thesis is that the approaches and lessons learned from modeling spatial relations will, to a large extent, generalize to other areas. Although, on the surface, spatial relations may seem intuitively simple to model, they have a subtle complexity that arises in the details of trying to build a computational model. Spatial relations are inherently vague, the boundaries of a spatial region are soft — perturbing an object slightly does not affect the spatial relations in which it participates. Most spatial relations are binary relations that hold between a pair of objects, which, unlike predicates of a single object, open the door to the recursive structure that is at the heart of the complexity of language. For instance, the recursive structure allows for arbitrarily long utterances: "the chair to the left of the desk that is behind the bed that is ..." It is for these reasons that we have chosen to focus our study on spatial relations.

1.1.1 Interpreting Utterances

In order to respond to a command or a query issued in natural language, it is necessary for a system to be able to interpret the utterance. The goal of interpreting an utterance is to translate a natural language representation of an utterance into a formal representation of the meaning contained within the utterance. In this thesis, we focus on interpreting utterances that refer to objects in a physical scene. For example, consider the scene in Figure 1.1.

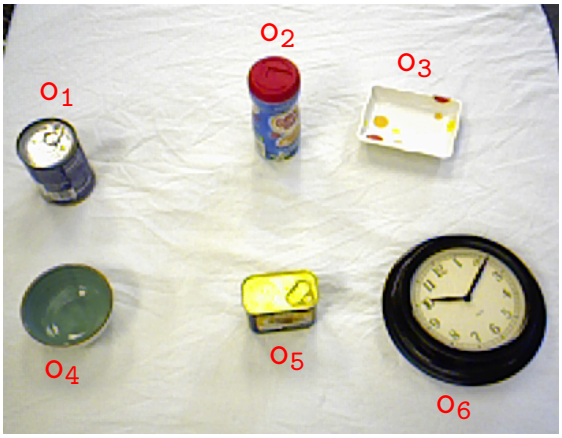


Figure 1.1: An example of the visual contexts used by our system to ground linguistic utterances.

The following natural language utterance evaluated with respect to Figure 1.1 refers to an object in the scene, o_2:

in the back behind the spam

A formal representation of the meaning of this utterance is shown in Figure 1.2.

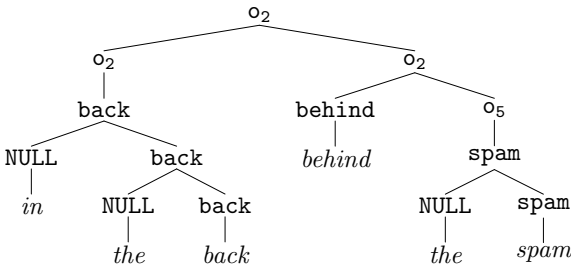


Figure 1.2: An example of the formal representation of the meaning of an utterance.

In order to construct this formal representation of the meaning from the natural language, the system must accurately model the lexical meaning of the words, and compose these meanings to construct the meaning of the sentence as a whole. We present a model of lexical semantics that learns the meaning of spatial relations (e.g. *behind*) from data. The meaning

learned for these relations is abstract, existing independently of any particular physical context, therefore it can be grounded in a new physical context to determine the relations that hold in a new scene. The learned model of lexical semantics is able to capture the subtlety in border cases, where a spatial relation gradually transitions from being absolutely true to being absolutely false, thereby capturing the vagueness in meaning present in spatial relations. Past work has focused on modeling the lexical meaning in isolation (Logan and Sadler, 1996; Regier and Carlson, 2001), or on modeling the meaning of entire sentences where the lexical meaning was grounded in a structured representation or virtual environment, as opposed to a physical environment with vague relations (Kate et al., 2005; Zettlemoyer and Collins, 2005). In contrast, we do both: model the lexical semantics of vague relations and combine the lexical meanings to compositionally define the meaning of the sentence as a whole. In this thesis, we focus on lexically modeling the meaning of nouns, unary, and binary spatial relations, setting aside meanings of adjectives and quantifiers. However, our model is robustly able to capture the variability of expression present in unconstrained natural language, and it fails gracefully by absorbing concepts that fall outside the scope.

1.1.2 Generating Utterances

In order to have a two-way interaction with a natural language processing system, the system must be able to generate utterances. We focus on the task of generating an expression that refers to an object in a complex scene because it is fundamental for any system that generates natural language. For example, consider the scene in Figure 1.3, where the task is to refer to the circled object, o_1.

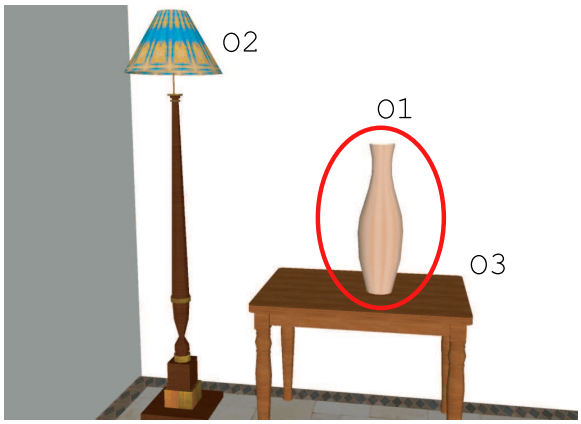


Figure 1.3: An example of a visual scene within which our system makes references to objects.

The possible output of the generation system includes: "right of o_2" and "on top of o_3."

The first of these two expressions is ambiguous, it can refer to either o_1 or o_3; whereas the second expression can only refer to o_1. Building a computational model that can represent this ambiguity and select the less ambiguous descriptions is the central challenge we address.

Our computational model relies on a model of vague spatial relations, distinguishing it from past work that generated expressions from a clear-cut database of facts (Dale and Reiter, 1995). To select the utterance that is less ambiguous, we model the generation process as a cooperative interaction between a speaker and listener that has been formalized as a mathematical game (Parikh, 1992; Benz et al., 2005). We focus our study on the selection of an appropriate spatial relation and reference object by allowing our model to refer to objects by IDs (e.g. o_1, o_2, o_3) or spatial expressions (e.g. *right of* o_2).

1.1.3 Deployment on a Robot Platform

To demonstrate the functioning of the interpretative module and the generative module, we deployed them on a robot platform, as is often done as a proof of concept (Tellex et al., 2011b). Ultimately, the test of whether a system was able to properly interpret an utterance is by seeing whether it can carry out an action as a result of understanding the utterance. We deployed the generation and interpretation systems on a PR2 robot that can respond to commands and queries.

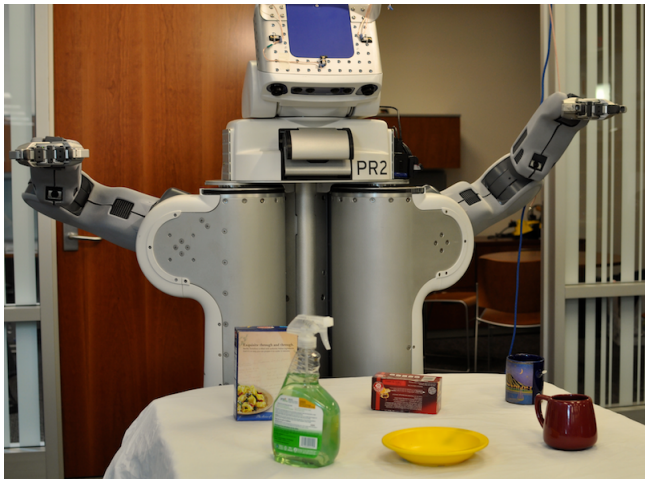


Figure 1.4: An example setting in which the PR2 robot is issued commands and asked queries.

The PR2 can pick up, put down, point at, and move objects as well as respond to queries about the locations of the objects. The key challenges were integrating the linguistic modules to work coherently with the robot and adding support for interpreting commands and queries built on top of the interpretation module.

1.2 Philosophical Background

The philosophical debate about the structure and function of language, meaning, and

truth is ongoing. This section presents the relevant background needed to appreciate how our modeling decisions are situated in the context of the deeper philosophical issues. Note that philosophy is an inspiration but not a focus of this work.

To address the nature of meaning in language one must answer the question of how language relates to reality. There have been two main approaches for answering this question that differ in their conception of natural language. The first approach was pioneered by Frege (1879), and it portrays natural language as a noisy representation of an underlying formal system operating under mathematical principles. The second approach follows the work of Austin (1962), and it describes natural language as a social activity, where speech acts are behavioral realizations of human intentions.

Colloquially, when asked to explain the meaning of a word, one might respond with a reference to an object in the world. For example, a natural answer to "What does 'the evening star' mean?" might be "'the evening star' means the planet Venus." In this regard, the meaning of a word is the object to which it refers; "the evening star" refers to the second planet from the sun, Venus.

Frege pointed out, however, that the sentence: "the evening star is Venus" means something different than: "Venus is Venus." The first sentence gives new information, whereas the second sentence does not. Frege argues that since this second sentence means something different than the first, there must be more to the meaning of "the evening star" than just the reference, Frege called this additional meaning the *sense*. The distinction between sense and reference is typically explored by examining *intensional* contexts, situations where the same word can have a different reference in each context. While the notion of sense is important, our task is referential in nature and does not generally involve sense distinctions of this sort. However, in our experiments, we found that a single object can be referred to in different ways. For instance, in the data we collect, O_2 in the scene presented in Figure 1.1 was referred to as *can of spam*, *canned meat*, *rectangular can*, *tin food*. One of the contributions of our system is a computational model that is capable of unifying these different senses as referring to the same referent.

Frege took the sentence as the primary object of meaning. The meaning of an expression (a linguistic unit smaller than a sentence) was derived from the meaning of the sentence in which it occurs. Frege formalized the meaning of a linguistic predicate — such as *baldness* — into a mathematical predicate. Mathematical predicates can be seen as mathematical functions that take an object as an argument and yield the value true if the predicate accurately describes the object. Equivalently, predicates can also be seen as mathematical sets of the objects that would make the predicate function true. The term "compositionality" refers to this idea that the meanings of the expressions within a sentence combine to give the meaning of the sentence as a whole. By analyzing the meaning of a sentence in terms of smaller modules, which can be rearranged in novel ways, Frege explained how we can attribute meaning to sentences we have not heard before. The notion of compositionality was a huge advancement towards explaining how language relates to reality. We build on Frege's formalization of the way that language works; we model nouns and unary spatial relations as predicates on objects and binary spatial relations as binary mathematical predicates. In order to employ probabilistic machinery to build a computational model of meaning, we extend

Frege's simple conception about the truth or falsity of an utterance with a probabilistic view that assigns a certain likelihood to an utterance being true.

One issue that arises with this simple referential model of meaning is the problem of vagueness, classically exemplified by Eubulides in the sorites paradox (aka "the paradox of the heap") (Hyde, 2011). The paradox concerns the vagueness inherent in any definition of a heap: a million grains of sand forms a heap; removing just one grain of sand from a heap results in a heap, albeit a slightly smaller one; repeatedly applying this reasoning would imply that a single grain of sand is a heap, leading to an apparent contradiction. Although this statement of the sorites paradox highlights the challenge of trying to define vague concepts such as a "heap," the problem also arises for predicates of a single object and relations between objects. The problem of vagueness arises when trying to model the meaning of gradable adjectives — those that can be true to varying degrees — such as those describing size (big, large, tall) and location (distant, remote). Of central relevance to this thesis, the problem of vagueness arises when trying to model the meaning of spatial relations; for example, if an object is located to the left of another, a single infinitesimal movement will not change the relation, but sufficiently many repeated movements will. Rather than resorting to partial degrees of truth (Zadeh, 1965), we take the view that a vague predicate is in reality either true or false (Sorensen, 1985), and we model the uncertainty about the objective truth with Bayesian probability.

Russell (1905) issued the strongest attack on Frege's work by looking at the case of definite descriptions. Russell's primary example was, "The present King of France is bald." This sentence is problematic; the description fails to refer to a specific object in the world because there is no present King of France. According to Frege, in order to interpret the sentence as being true or false, one must resolve the reference of the subject of the sentence. However, since the subject fails to refer, the sentence cannot be interpreted, and hence it cannot be said to be either true or false. Although formally the subject seems to fail to refer, this sentence can be understood on an intuitive level, but this intuition falls outside Frege's theory of meaning. Russell proposed an analysis of this sentence which does have a truth value. Russell's analysis of the sentence can be paraphrased as: "There exists a unique entity x such that x is the present King of France and x is bald." Because no such entity exists, under Russell's analysis, the sentence is false.

Russell wanted a theory of meaning that completely characterized the set of interpretable sentences. In his desired theory, Russell wanted a formal theory that could logically express all sentences that can be said to be true or false; and cannot express sentences that have no truth value. Hence the distinction between meaningful and nonsensical sentences would be reduced to a syntactic decision about what is expressible. In order to accomplish this goal, he ended up having to invent a theory of types to prevent self-contradictory descriptions ("the set of sets that does not contain itself") from being logically expressible.

Despite the disagreements, Frege's notion of a presupposition failure and Russell's notion of nonsensical sentences, each in their own way, limit the scope of the enterprise of analyzing natural language in terms of formal logic. They both claim that some sentences simply cannot be evaluated as being true or false. The models presented in this thesis are not rich enough to adequately distinguish the common utterance from one with a presupposition

failure or nonsensical expression. Our models treat all utterances identically and return an interpretation regardless of whether the sentence can be evaluated as being true or false.

Putting aside the peculiar cases of definite descriptions that fail to refer, Austin points out that there are many sentences that arise naturally that cannot be interpreted as true or false. Austin distinguishes constative utterances — those which make a claim about the state of affairs in the world — with performative utterances — which perform an action through speech. Unlike constative utterances, performatives cannot be said to be true or false, and so their meaning cannot be understood in terms of their conditions of truth. Promises, commands, declarations are examples of sentences that fall in this category. Austin introduced the concept of a speech act, which, similar to Wittgenstein's conception of "language as a tool," (Wittgenstein, 1953), explored characterizations of meaning of language in terms of its use. Austin claims that constative sentences are a special type of speech act whose intended action is to convey information about the world. The speech act functions as a wrapper around a proposition, specifying how one should process the contained proposition. For example, a command encourages the hearer of an utterance to make the proposition true, and a query encourages the hearer to provide the missing piece of information. We deploy our models on a PR2 robot that is capable of responding to commands and queries, thereby, to a certain extent, engaging in a speech act.

Similar to Austin, Grice (1975) sought to characterize the meaning of language at the level of an interaction between a speaker and a listener rather than at the level of the proposition, as Frege did. Grice pointed out that often the speaker's intended meaning deviates from the literal linguistic meaning. For example, the literal interpretation of the utterance, "I am out of petrol" would simply be a statement about the world; however, it is natural to interpret the speaker's intended meaning as additionally requesting directions to the nearest petrol dispenser. Grice uses the term conversational implicature to refer to the component of meaning implied by the utterance that is not literally stated in the utterance. Grice presents a set of conversational maxims, which together form a cooperative principle that characterizes how interlocutors derive conversational implicatures from discourse. The algorithm our system uses to generate a referring expression that is unambiguous is a computational realization of some of Grice's maxims.

Taking the philosophy as inspiration, we engineered a robotic system that can interact with humans through natural language. Along the way, we had to present a solution to the challenge of relating language to reality. Our solution is motivated by engineering principles: a linguistic reference to an object in the world grounds into the output of an object recognition system, and whether a relation holds between two objects is determined by a computational model of spatial prepositions. For the sake of expediency, we have simplified the complex issues of presupposition failure and the subtle distinctions between sense and reference. We present a system that can engage in a certain speech acts and interpret linguistic utterances in the domain of spatial relations, even when the utterances contain vague concepts. Although we have made progress towards building an artificial agent that can manipulate language, there are many avenues of future development.

1.3 Spatial Semantics Background

In this section, first we will give a survey of the standard terminology that we will adhere to throughout the course of this thesis. Then we will summarize the notable studies in Linguistics that have influenced the work on computational models of spatial semantics.

1.3.1 Terminology

There have been many books, special issues of journals, monographs, and articles exploring the topic of spatial semantics (Zlatev, 2007). In a thoroughly researched survey chapter, Zlatev (2007) summarizes the salient concepts common to nearly all studies on spatial semantics. They are a comprehensive and minimal set of concepts needed to represent the meaning of any spatial description. These defining concepts are best explained by means of an example. Consider the following spatial description:

The car is behind the house.

The relevant concepts are:

- The *target* (*trajector*, *figure*, *referent*) is the entity about which a spatial expression is made. The *trajector* is "the entity whose (trans)location is of relevance" (Zlatev, 2007).

  In the example, "the car" is the target.

  In this thesis, I use the term *target* or *target object* because the applications presented here involve selecting the target from amongst a set of objects. Depending on the presence of motion, the target can be *dynamic* or *static*. In this thesis, I focus only on static targets.

- The *reference* (*landmark*, *ground*, *relatum*) is the entity in relation to which the location of the target is defined.

  In the example, "the house" is the reference.

- When conceptualizing space in terms of coordinate geometry, the *frame of reference* (*perspective*, *perspective system*) can be thought of as the coordinate axis that fixes the direction for spatial relations. Zlatev (2007) distinguishes three frames of reference: *intrinsic* (object centric), defined in terms of the orientation of the reference object; *relative* (viewpoint centric), defined in terms of the speaker/listener's viewpoint; *absolute* (geocentric), defined in terms of a fixed global orientation (e.g. "north").

  A *projective* relation is a spatial relation defined in terms of a specific direction along a frame of reference (e.g. "to the left of," "above," "to the north of"). Projective relations are distinguished from *topological* relations that pertain to containment or distance (e.g. "inside of", "near to").

A house has an intrinsic frame of reference (cf. "front door", "backyard"), and therefore the example is ambiguous as to whether the projective relation "behind" is defined with respect to an intrinsic frame of reference or relative frame of reference. If the expression is understood with respect to an intrinsic frame of reference, the example can be paraphrased as, for example, "The car is located in the backyard of the house." In the other interpretation, the example might be uttered by a speaker located in the backyard of the house describing the location of a car in the front-yard from a relative frame of reference.

In this thesis, we do not explicitly model the distinction between these different frames of reference and likely conflate intrinsic and relative the meanings of spatial expressions. We note that the data consists of photographs of indoor scenes, which means that the speaker describing a scene and the listener interpreting the description both perceive the scene from the same viewpoint — obviating the need to model any distinctions between speaker and listener frames of reference.

The remaining concepts presented in Zlatev (2007) pertain of the path, direction, and motion of dynamic targets, which are outside the focus of this thesis.

1.3.2 Linguistic Models of Spatial Expressions

When building a new model of spatial relations, exploring the history of study in the field will prove insightful. The linguistic studies of spatial relations broadly highlight the challenges that arise when modeling spatial relations. The emphasis of these linguistic studies often focuses more on intuitive judgments and less on computational complexity or representability. We summarize the findings of these studies here because they provide inspiration of the computational models we will present later.

The classical model of how spatial relations (and prepositions in particular) convey meaning in language was expressed in the seminal work of Herskovits (1987). Herskovits (1987) argues that prepositions have a literal meaning that stands independent of any particular expression. For example, the literal meaning of "X is on Y" is expressed as "X is contiguous with Y, and Y supports X." The argument continues that the literal meaning of prepositions is inadequate to characterize the full scope of their use in language. Instead, the meaning of a preposition adapts depending on the context in which it is used according to either conventional influences (e.g. "the wrinkles *on* his forehead"), or approximations to the literal meaning (e.g. "the village *on* the road to London"). Although Herskovits (1987) is influential in characterizing the challenges present in modeling the meaning of spatial relations, the proposed model of meaning is unsatisfying in that they do not readily lend themselves to a computational approach. For instance, it is debatable whether the notion of "supports" in the definition of "on" is any more interpretable than the notion of "on" itself.

Following Herskovits (1987), Landau and Jackendoff (1993) provide an appreciable list of the spatial prepositions of English and postulate common defining features of their meaning.

The salient features affecting meaning include the aspects of target and reference object geometry, the relation of the region in relation to the reference object, the choice of axis system, and whether the target object is visible or occluded. These features are generally defined in terms of measurable quantities of the target and reference objects: relative position in space, shape, presence or absence of contact, and distance. Although Landau and Jackendoff (1993) do not provide a computational model, the work makes significant progress in that direction. Indeed, we take as inspiration some of the results presented in this study when defining the features of our model of spatial prepositions.

Several works have proposed linguistic explanations of the meaning of spatial expressions by means of compositional interpretation (Nam, 1995; O'Keefe, 1996; Zwarts and Winter, 2000). In particular, Zwarts and Winter (2000) present a vector space model of the meaning of locative phrases as well as the modifier phrases that can affect the meaning of these locative phrases. The interpretation of a locative expression is captured with a vector space formalism by algebraically manipulating sets of vectors and regions of space. For example, the interpretation of "ten meters outside the house" is an intersection of the set of vectors that are ten meters long with the set of vectors that start inside the house and end outside. The meaning of each preposition in the model is simple and intuitive. For instance, the meaning of "right of X" is a quadrant of space (disjoint from "left of X", "above X", "below X") that is constructed from the set of vectors whose projection along a vector starting at X and extending to the right is larger than the projection along a perpendicular vector. Although the compositional power of the model to explain the meaning of modified locative expressions is compelling, it is not sufficiently evaluated against empirical evidence. At present, our models of spatial prepositions and compositional semantics do not support modified locative expressions; however, Zwarts and Winter (2000) would be the appropriate starting point for extensions covering this topic.

1.4 Computational Models of Spatial Relations

Having developed some linguistic and philosophical intuition about the semantics of spatial relations, we now consider several computational models that make this intuition precise. We consider how to automate the task of judging how well a spatial relation describes a scenario using computational models that are accurate and flexible enough to capture the variance inherent in spatial relations.

1.4.1 Challenges

Space is continuous, and language is discrete; the fundamental challenge in computationally modeling the semantics of spatial relations has been in bridging this gap. At what distance does "near" become "far"? At what angle does "X is to the left of Y" become

"X is in front of Y"? The computational approach attempts to quantitatively answer these questions about the vagueness of spatial relations.

Most computational approaches to modeling spatial relations have decomposed the problem into the modeling of topological and the modeling of projective spatial relations. The challenges that arise when modeling the meaning of topological relations pertain to the effect of the distance between the target and reference objects. Models of projective relations focus on how the angle between the target and reference objects affects meaning as well as resolving which frame of reference is active in a given spatial expression.

The main lines of inquiry (summarized below), as well as our contributions, ignore the epistemological doubt that the meaning of spatial relations can be formalized into a computational model, generally presented by Dreyfus (1972) and exemplified in the context of spatial relations by Herskovits (1987). For example, Herskovits (1987) argues that to interpret, "The bird is *in* the bush," requires computing the gestalt closure of the bush; and "The bulb is *in* the socket," can only be felicitously claimed by someone with a functional understanding of light bulbs and light sockets. Arguably, these examples require a holistic understanding of an extremely broad background knowledge that does not readily lend itself to computation (Lenat et al., 1985). Rather, the prior work summarized below as well as our computational models have prioritized accurate and computationally feasible modeling of spatial relations before addressing these challenges.

1.4.2 Models

A quantitative model of a binary spatial relation scores a triple consisting of a target object, a preposition, and a reference object. There are several options for how to compute the score of the triple, which have been explored in the prior work. Historically, the two main lines of research into computational modeling of spatial relations have been the "binned non-parametric" template based methods and the parametric potential function based methods. The template based methods approximate continuous space by discretizing it into grid cells centered around the reference object; the assigned score depends on the given relation and which grid cell the target object occupies (Hayward and Tarr, 1995; Logan and Sadler, 1996). Taking inspiration from physics, potential function based methods instantiate a potential field for a given relation and reference object; the field assigns a score to points in space (idealized target objects) depending on their location (Yamada et al., 1988). Our approach is more similar to the parametric based methods, however rather than assigning a score to points in space, our models assign a score to the bounding box of the target object.

Within the category of binned non-parametric methods, Logan and Sadler (1996) introduce the notion of "spatial template" to mean a representation centered around the reference object that is aligned with the reference object's intrinsic frame of reference and defines regions of acceptability. The degree to which a spatial relation is acceptable depends on which region the target object occupies in the spatial template. Fuhr et al. (1995) model the region of acceptability as open regions of space specified to extend infinitely from the boundaries of the bounding box of the reference object. However, the regions in Fuhr et al. (1995)

are coarse and can contain many objects; the model can not represent gradations in human judgment about the acceptability of two objects that occupy the same region. Addressing this point, Logan and Sadler (1996) explore a finer grained discretization of the space. They model the region of acceptability as a 7×7 Cartesian grid centered on the reference object and use human judgments as data to empirically estimate acceptability of these regions for various topological and projective prepositions. Carlson-Radvansky and Logan (1997) generalize the approach of Logan and Sadler (1996) to handle variation in reference frame. In all these template based methods, the coarseness of the discretization directly determines the model's resolution of spatial judgments. Although decreasing the coarseness would result in more nuanced judgments, the corresponding increase in resolution would also lead to data sparsity.

Yamada et al. (1988) introduces the notion of a "potential function" that captures spatial judgments with a parametric function of the angle and distance between the target and reference objects. Unlike the spatial template based methods, potential function methods separately model topological and projective prepositions due to the challenge of uniformly parameterizing the meanings of these distinct types of prepositions. Yamada et al. (1988) proposes distance potential functions to model topological relations, directional potential functions to model soft projective relation judgments, and inhibited half-plane potential functions to model hard projective judgments. These potential functions are intuitive and take inspiration from physics, however they contain free parameters which were manually chosen. Gapp (1995) models projective prepositions with a simple function of the angle between the target object and the reference object; the predictions of acceptability of the model are compared against human judgments. Regier and Carlson (2001) look at how the size of the reference object affects human judgments of acceptability of a projective relation. Their results show that integrating the angular judgments over the entire volume of the reference object accurately captures human judgments. By looking at functional objects rather than geometric shapes (as in the previous studies), Coventry et al. (2005) show how the function of an object affects human judgments of projective relations. Costello and Kelleher (2006) use simple models of distance and salience to study the influence of a distractor object (an object distinct from the target or reference) on human judgments of nearness.

We propose a single set of features that we use to learn a parametric model of projective and topological prepositions that is trained based on human judgements. Although we do engineer different sets of features with the intention of modeling different types of prepositions, we let the learning algorithm determine how to assign weights to these features for modeling each preposition. The models presented in this thesis focus on the fundamental issue of representing the meaning of a preposition without explicitly considering the effects of object function or frame of reference. Our models are structured to pick a target object given a preposition and a reference object, thereby implicitly capturing the effects of distractor objects on preposition meaning.

1.5 Natural Language Understanding

Natural language understanding (NLU) is the task of automatically interpreting a query issued in natural language in order to generate a meaningful response about a knowledge base. We focus on the task of automatically interpreting a reference to an object in a visual context that is expressed in terms of spatial relations. Although there has been some work on directly predicting the response from the words in the query (Vogel and Jurafsky, 2010; Branavan et al., 2009), most approaches first construct a meaning representation of the query prior to generating the response. The meaning representation is typically an expression written in a formal language such as Prolog or lambda calculus. We propose a context free grammar uniquely designed to represent spatial references to objects, similar to the semantic context free grammar presented in Börschinger et al. (2011).

Most approaches train an NLU system from natural language queries annotated with their meaning representation. The core challenge when learning an NLU system from labeled data is the word alignment problem, which is the challenge of determining the alignment between words in the query with the corresponding fragments of the meaning representation. Kate et al. (2005) present a transformation based system that learns a set of rules that convert the natural language query into a meaning representation. Wong and Mooney (2006) treat the problem as a string-to-tree machine translation task; they use word alignment to learn synchronous context free grammar that produces the natural language query on one side and a meaning representation on the other. Kate and Mooney (2006) learn a probabilistic context free grammar whose terminals are the words in the natural language query and whose internal nodes correspond to the meaning representation; the system uses SVMs to estimate the weights of production rules and address the word alignment problem by searching over all possible partitions of the input sentence. Ge and Mooney (2005) learn a semantic parser from semantic parse trees fully annotated with the word-concept alignments. Lu et al. (2008) present a hybrid tree approach that has production rules with words as well as non-terminals on the right hand side; the model treats the alignment between words and concepts as a hidden variable and uses the inside-outside algorithm to estimate parameters. The aforementioned work keeps the grammar of the meaning representation fixed and treats the NLU task as the task of parsing a natural language utterance into this fixed semantic grammar. In these approaches, the semantic grammar is rich; it specifies the set of predicates that exist and the predicate-argument structure. In contrast, Zettlemoyer and Collins (2005) choose to use the combinatory categorial grammar (CCG) (Steedman, 1990) formalism. The CCG formalism pushes all of the complexity into the lexicon, requiring only a few simple and domain independent combination rules for parsing.

All of the approaches above require collecting a large set of annotated training examples, which can be a costly undertaking. To ease the burden of collecting data, we present a model that learns an NLU from partial annotation, where the majority of the meaning representation is not present during training. Several other studies have explored building NLU systems from less annotation (Clarke et al., 2010; Liang et al., 2011). Similar to our work, these systems are trained from pairs of natural language query and response, while

treating the meaning representation as unobserved. However, these systems are grounded in a database of facts that are either true or false, the issue of vague relations is avoided.

Most applications of NLU are a natural language interface for querying a knowledge base represented as a database; however, some studies have considered applying NLU to knowledge bases in a different format. These alternate formats include structured representations such as: symbolic transcriptions of RoboCup sportscasts (Kim and Mooney, 2010; Chen et al., 2010; Börschinger et al., 2011); manuals explaining how a computer game should be played (Branavan et al., 2011); and chat conversations between between a customer and a customer service representative pertaining to airline reservations (Artzi and Zettlemoyer, 2011). We use the physical world as our knowledge base, which implicitly contains relations between objects. The relations are not symbolically represented, as they would be in a database, but instead we have to access them through perception, which introduces a layer of noise that is not present in these previous studies.

1.6 Applications

For decades, there have been many explorations into engineering an artificial agent that can interact with a human via natural language to accomplish a goal in a given environment. To have a rich interaction, the agents must interpret commands and queries and respond appropriately via action or by answering via generating a linguistic utterance. These agents understand language by grounding it in a model of the environment, they represent the meaning of linguistic commands and queries in terms of the arrangement of objects in the world. Historically, the grounded semantics systems have modeled language of varying complexity and models of the world of varying richness. The complexity of worlds varied in terms of complexity of the space and the variety and properties of the objects found within the space. The complexity of space has ranged from simple discrete environments where each location is specified as a cell in a grid to virtual environments where an object can occupy a continuum of locations. Recently, several studies have explored situating the artificial agents in the real world where objects can take any configuration that respects the physical laws of the universe. Moreover, agents operating in the physical domain are burdened with the challenge of determining the position of objects through perception. Related to the complexity of space comes the variety of objects that occupy that space. In virtual environments, the complexity of objects has ranged from simple blocks and geometric shapes to more complex $3D$ models of real, physical objects. In the physical environments, objects have nuanced properties and vary in appearance; a robust artificial agent needs to recognize the physical objects with a computer vision system. In this section, we explore where prior research has dealt with these issues of complexity when interpreting and generating spatial descriptions to objects in the world.

One of the first significant applications of a grounded language system was SHRDLU (Winograd, 1972). The system was a simulated robot that could manipulate blocks in response to commands and answer questions pertaining to the locations of the blocks via

natural language. SHRDLU was a compelling demonstration of a grounded natural language system that outlined the key challenges and motivated further study to advancing the capabilities of grounded natural language systems. The natural language systems prior to SHRDLU were limited in various ways. For instance, ELIZA (Weizenbaum, 1966) could only perform syntactic transformations on its input; it did not have a rich representation of the meaning of a sentence. There were specialized systems, such as linear algebra systems (Bobrow, 1964), which would only be able to solve a problem that fell inside their domain. As the form of the meaning representation became more general, converting natural language into this internal representation became more complex. SHRDLU addressed these challenges by grounding the meaning in a simulated world that had the right amount of complexity to logically manipulate meanings while still being simple enough to reach from natural language.

The main limitations of SHRDLU arose from the limits of the virtual environment and implementation of natural language understanding. The environment did not contain any complex objects with complex properties (only blocks of various colors and shapes), and space was represented as a finite discretized grid. Converting the natural language to representations in terms of the virtual environment was done with routines specified with significant manual effort. The meanings of the spatial relations (*located at*, *support*, *right*, *left*, *behind*, *front*, *above*, *below*) were hard coded and brittle, and extending the system to broaden the coverage would require additional manual effort. SHRDLU was a remarkable sketch of the challenges present in building a grounded natural language system, and it outlined the avenues for further research. Below, we present the recent work on two of the subproblems highlighted by SHRDLU that are relevant for this thesis: interpreting and generating spatial descriptions.

1.6.1 Interpreting Spatial Descriptions

Below we summarize the previous systems that relate the meaning of spatial descriptions to the configuration of objects in the world. All of these systems limit their scope by restricting their representation of reality (e.g. focusing on a virtual environment) and/or focusing their models on a subset of language (e.g. sequences of simple descriptions rather than complex nested descriptions). Each system described below uses a different structured representation to ground the meaning of the natural language depending on the application. The systems are related to our models in that they aim to interpret spatial descriptions presented in natural language; however, they differ in application, technique, and coverage of linguistic complexity from the models presented in this thesis.

Siskind (1994) focuses on interpreting animations of a stick figure manipulating a ball. Rather than directly grounding linguistic descriptions into the configuration of the world, the work focuses on describing the configuration of objects in the video in terms of logical relations, with the intention of ultimately deducing which linguistic relations hold by means of these relations. The model computes the truth of primitive relations (such as an object's position or a support relation between objects) in the scene by segmenting and computing

approximate change across frames in the animation. The truth of some of these primitive relations is determined by counterfactual simulations (e.g. "the ball would have fallen through the table if the table were not supporting it"). From the truth of these primitives, the model logically deduces the truth of complex relations such as whether an object is being thrown, falling, or rolling. The meaning of the complex relations are manually defined in terms of the primitive relations by means of logical expressions. However, the mapping from natural language utterances to logical forms is complex and context dependent, which is not demonstrated in this work; limiting its usefulness in interpreting spatial descriptions. The main differences from our work are the focus on dynamic, rather than static scenes, the deterministic logical reasoning approach, rather than our empirical and statistical approach, and the emphasis on constructing a logical representation of the world rather than the focus on interpreting real linguistic utterances.

To explore the interpretation of more complex linguistic expressions, Gorniak and Roy (2004) consider the task of interpreting references to objects in a virtual environment restricted to an arrangement of green and purple cones on a flat surface. Their model uses a context free grammar to represent the meaning of a referring expression and to organize the interpretation of the expression. The model supports interpreting concepts of color, spatial extrema, and reference to groups of objects; in contrast, our system does not interpret these concepts. Gorniak and Roy (2004) use the AVS model (Regier and Carlson, 2001) to interpret projective prepositions, and a hard coded function for topological prepositions. Both their system and ours compute the target object referred to by a referring expression that could possibly contain multiple spatial relations. and return the target object to which they refer. Aside from the particulars of the linguistic formalism and model of spatial relations, our system differs from theirs in that we interpret expressions in physical contexts, whereas they focus on the restricted virtual environments.

Moving to a richer model of the world, Kelleher and Costello (2009) describe how to incorporate models of topological and projective prepositions into a dialog system that can interpret spatial descriptions in physical scenes. Similar to our method, their system parses an utterances, extracts the referring expressions contained in the resulting parse, and computes the point in space or target object being referred to by the referring expression. Although their system appears to be able to understand references to objects in the physical (as opposed to virtual) world, Kelleher and Costello (2009) do not fully specify how their system handles complex language, such as descriptions containing multiple referring expressions; namely, it does not specify how the interpretations of these multiple referring expressions are combined. Furthermore, the focus of the empirical results is on the performance of the prepositional models rather than the linguistic coverage and interpretation accuracy. In contrast, we evaluate our model on the accuracy of the interpretation of utterances that can contain multiple referring expressions.

Tellex et al. (2011b) present a model that is able to interpret complex linguistic expressions that refer to events, objects, and relations in the physical world. Their model is evaluated on the following tasks: indoor navigation, spatial language video retrieval, and mobile manipulation. The model parses a natural language utterance, and, conditioned on this parse, breaks the problem of interpreting the utterance into independent subproblems

of interpreting the subexpressions present in the parse. Each subexpression can refer to an event or command, an object, a location, or a path. The parse of a given sentence instantiates a locally normalized graphical model, and the parameters of this graphical model that govern the interpretation of each subexpression are learned from supervised training data. In contrast, our model does not require full labeled linguistic structures to train, but instead is learned from partially annotated data consisting of a natural language sentence paired with the target object to which it refers. Not requiring a full labeled linguistic structure lightens the burden of gathering linguistic annotation and more closely resembles the method by which children learn language.

There have been several related studies that, rather than focusing on static prepositions, have focused on modeling dynamic spatial descriptions by exploring models that follow directions to navigate from an origin location to a destination on a map (MacMahon et al., 2006; Levit and Roy, 2007; Vogel and Jurafsky, 2010; Tellex, 2010; Chen and Mooney, 2011). These models ground the meaning of language in terms of a path through a virtual or physical environment. To follow a set of directions to a destination, these systems must perform the following key steps: interpret references to landmark objects, understand dynamic prepositions describing path segments (e.g. "around the graveyard"), disambiguate between reference frames, and compose the path segments from each individual direction into a continuous path. The linguistic nature of directions tends to be a relatively flat sequence of steps, rather than the deeper, nested descriptions typically found in references with static prepositions. The models of navigation planning differ in focus from our models because they primarily focus on modeling dynamic rather than static prepositions, and they use a sequence model for representing descriptions rather than a tree model which can capture the recursive structure of static spatial reference.

Taking a completely new perspective, Coyne and Sproat (2001) and Zitnick and Parikh (2013) present systems that convert a textual description of a static scene into a rendering of the scene. These systems are fundamentally different from both our model and the previously mentioned work in that they *output* an environment, rather than interpret natural language with respect to a given, fixed environment. Specifically, the system presented in Coyne and Sproat (2001) parses each sentence in the description using a dependency parser, the target and reference objects participating in each spatial relation are extracted, and the relation is converted into a constraint on the location of the objects. Most objects contain certain spatial regions that determine how they interact with other objects. For instance, the description, "The daisy is in the test tube," can apply to a scenario where only the stem of the daisy is contained in the test tube even if the petals are outside of the test tube. To account for complex interactions of this sort, regions of the objects are annotated with spatial tags (e.g. the stem of the flower is annotated with a *stem* tag), which the system uses to generate the spatial constraints. The association of words to objects in the $3D$ object database, the spatial tags on the objects, and the generation of constraints are either manually specified or determined by hard-coded rules. The entire scene is rendered by selecting all target and reference objects from a database of $3D$ models, and arranging them to satisfy the spatial constraints.

1.6.2 Generating Spatial Descriptions

Natural language generation (NLG) is the inverse task of interpretation, where the system starts with a meaning representation and must produce a natural language utterance. According to the thorough survey presented in Krahmer and Van Deemter (2012), referring expression generation (REG) is a critical component of nearly any NLG system, and for this reason, we focus on REG in our work on generating spatial descriptions. REG is the task of producing a natural language description of an object in a context so that a hearer can identify the target object from among a set of distractor objects (Reiter and Dale, 2000). Often, there are many possible descriptions of the target object, and the central challenge of REG is determining which a description is best. As Grice (1975) demonstrates, many utterances have conversational implicatures beyond their literal meaning. In the case of REG, an incorrect implicature might result in a hearer misidentifying the target object, thereby picking a distractor object instead. To reduce unintended implicatures, Grice (1975) claims that speakers follow the following maxims:

- maxim of quality: be truthful
- maxim of quantity: provide exactly the sufficient amount of details — no more, no less
- maxim of relevance: be relevant
- maxim of manner: be clear, unambiguous, and brief

Most REG algorithms have been designed to the specifications established by the Gricean maxims. We design our REG as a computational realization of Grice's maxims of quality and manner.

Dale and Reiter (1995) consider the task of referring to objects by generating expressions that are unambiguous and brief. In their setup, each object is characterized by a set of attribute-value pairs, and an attribute-value pair does not uniquely identify an object. The problem of generating a brief yet unambiguous referring expression is formalized as choosing a small set of attribute-value pairs that uniquely determine the target object. Finding the smallest such set is NP-hard, so Dale and Reiter (1995) present several approximation algorithms. This attribute-value representation is a simplified model of reference; each word is a binary function over objects, which, unlike in our model, precludes expressing gradations in meaning or relations between multiple objects. Furthermore, Rohde et al. (2012) show that occasionally ambiguous words are preferred over more costly alternatives, highlighting the complex interplay between ambiguity and brevity.

When generating a referring expression, a speaker seeking to minimize the chance of a misunderstanding will consider the ways in which a listener might interpret his utterance. Similarly, a careful listener may consider the speaker decisions when interpreting the utterance. The strategic reasoning that a speaker and a listener make about each other's understanding of the meaning of an utterance is naturally modeled by a mathematical game (Parikh, 1992; Benz et al., 2005). Empirical results suggest that human speakers and listeners naturally reason about the "best response" of their interlocutor (and occasionally

recursively consider how the interlocutor reasons about them) when generating and interpreting a referring expression (Frank and Goodman, 2012; Degen et al., 2012). Similarly, our REG system explicitly models the ways in which potential listeners might misunderstand a given utterance, and select the utterance that reduces the likelihood of misunderstanding.

The aforementioned models of REG only support generating expressions containing simple predicates of a single object (e.g. names, colors, size), they do not allow the generated expressions to contain relations that identify the target object by relating it to other objects in the context. Dale and Haddock (1991) present an algorithm that can generate descriptions using relations beyond simple predicates; they note that their algorithm suffers from the "recursion problem." The recursion problem refers to the infinite regress that can result when describing a target object in terms of a reference object; namely, the reference object can further be described in terms of the target object thereby introducing a cycle which can lead to redundant descriptions of arbitrary depth. Gardent (2002) addresses the recursion problem by formulating the task of REG with relations as a constraint satisfaction problem with the objective to minimize the total length of a description, preventing the generation of redundant utterances. To a similar end, Varges (2004) formulate the REG task with relations as a graph search, where each search state is a description that can be expanded by elaborating the description by paying a cost. This graph search formulation supports the application of arbitrary search algorithms to find the lowest cost description, which again avoid the recursion problem. In our work, we focus exclusively on reference to objects via relations, and we avoid the recursion problem by limiting the maximum length of a referring expression that can be generated by our system. Furthermore, our models refer to objects via spatial relations, which are inherently vague. In contrast, all the aforementioned former work relies on a knowledge base of relations (or equivalently, a graph representation of relations (Krahmer et al., 2003)), which are fully observed boolean relations that either hold or do not hold of a set of objects — they do not address the challenge of referring with vague or graded relations.

At present, there have been relatively few studies into REG systems that operate over vague relations. A notable exception is Kelleher and Costello (2009) who adapt the REG system of Dale and Reiter (1995) to generate expressions that refer to objects in a visual scene via referring expressions containing spatial relations. To derive the set of boolean spatial relations needed by the underlying REG algorithm, Kelleher and Costello (2009) first apply their computational model of spatial relations to the objects in the scene and then keep only the relations where the score surpasses a fixed threshold. Although not strictly REG, several systems have been presented that generate a description of an entire image, rather than a single entity within the image (Farhadi et al., 2010; Kulkarni et al., 2011; Yang et al., 2011; Ordonez et al., 2011; Mitchell et al., 2012). In particular, Kulkarni et al. (2011) and Mitchell et al. (2012) use hard-coded models of pixel configurations to determine the preposition relations to generate, Yang et al. (2011) only generates functional (not spatial) prepositions based on word distributions in a large corpus, whereas Farhadi et al. (2010) and Ordonez et al. (2011) do not focus on prepositions in particular.

Chapter 2

A Grounded Computational Model of Spatial Relations

A computational model of spatial judgments is a necessary component for any grounded semantics system that must manipulate the meaning of spatial relations based on perceptual cues. Indeed, we use such a model to interpret (Chapter 3) and generate (Chapter 4) spatial descriptions in isolation as well as in the context of a human-robot interaction (Chapter 5). The prior work on modeling spatial relations separately modeled topological and projective relations (Logan and Sadler, 1996; Gapp, 1995; Regier and Carlson, 2001); instead, we present a coherent model of spatial relations that learns both topological as well as projective relations using a single set of features. Using a machine learning approach, our model learns from data to discern the combination of features that are relevant for each spatial relation. We focus our study on binary spatial relations, specifically the following projective and topological prepositions: *left of*, *right of*, *above*, *below*, *in front of*, *behind*, *on*, *under*, and *across from*.[1] In this chapter, we present the method of data collection, the machine learning features, and the training procedure that we used to develop a probabilistic model of spatial relations that lends itself well to integration into more complex systems.

2.1 Data Collection

In order to train our model, we collect human judgements about the spatial relations that hold in a particular $3D$ visual context. We annotate 43 visual contexts from the Google Sketchup $3D$ Warehouse, each containing an average of 22 objects (household items and

[1]This set of prepositions are the ones most commonly used by people to describe objects in a preliminary data gathering experiment.

Figure 2.1: An example of one of the Google Sketchup 3D models used in data collection.

pieces of furniture arranged in a natural configuration). These visual contexts are $3D$ computer models, which allows us to focus on modeling the meaning of the spatial relations in terms of the object configurations, while treating the tasks of object segmentation and determining the $3D$ position of the objects as given. A representative example of one of these visual contexts is given in Figure 2.1.

We collected annotations of these $3D$ models by using Amazon Mechanical Turk (MTurk). The human annotations for a particular context $\mathcal{C}$ take the form of triples $(o, w.r, w.o)$, where target object o is judged to be in relation $w.r$ to reference object $w.o$. For example, the triple $(\mathsf{o}_1, \textit{right of}, \mathsf{o}_2)$ captures the human judgement that object o_1 is located to the right of object o_2.

The interface for our MTurk task is shown in Figure 2.2. In this task, human annotators generate spatial descriptions. They are prompted with a target object o and asked to each produce an utterance w (by selecting a preposition $w.r$ from a dropdown list and clicking on a reference object $w.o$) that best informs a listener of the identity of the target object. For each object o in a visual context, we collect 3 descriptions, resulting in a total of 2,860 annotations, which we randomly partition into equal sized training and testing sets.

2.2 A Log-Linear Model of Spatial Relations

A computational model of binary spatial relations assigns a score to a triple consisting of a target object, a spatial relation, and a reference object within a visual context. We choose to compute this score using a log-linear model because it is a well defined probabilistic object with properties that facilitate incorporating it as a component in a larger system.

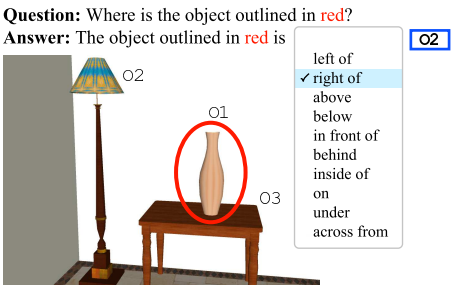


Figure 2.2: Mechanical Turk speaker task: Given the target object (e.g., o_1), a human speaker must choose an utterance to describe the object (e.g., right of o_2).

The structure of log-linear models allows the parameters to be easily fit to data. In the field of Natural Language Processing, log-linear models have become one of the standard methods to define a score in terms of a set of features.

2.2.1 Log-Linear Models

We use a log-linear model (aka maximum entropy model), that takes the following form:

$$p(y|x,\mathcal{C},\theta) = \frac{\exp \theta^T \phi(x,y,\mathcal{C})}{\sum_{y' \in \mathcal{C}} \exp \theta^T \phi(x,y',\mathcal{C})}$$

where θ is a real vector containing the parameters of the model, and $\phi(x,y,\mathcal{C})$ is a feature vector computed from the input x, output y, and context $\mathcal{C}$.

Given a set of annotated data $\mathcal{D} = \{(x_i, y_i, \mathcal{C}_i)\}_{i=1}^N$, we can define the log-likelihood of the data under the model as a function the parameters θ:

$$\mathcal{L}(\mathcal{D},\theta) = \sum_{(x,y,\mathcal{C}) \in \mathcal{D}} \log p(y|x,\mathcal{C},\theta) \tag{2.1}$$

A common approach for training a log-linear model from a given annotated dataset is to find the parameters that maximize the log-likelihood of the data:

$$\theta^* = \arg\max_\theta \mathcal{L}(\mathcal{D},\theta)$$

This optimization is typically performed using a gradient-based method such as gradient-ascent or L-BFGS; in our experiments, we use L-BFGS. These gradient-based optimization methods require both the objective function (2.1) as well as the gradient (2.2):

$$\nabla_\theta \mathcal{L}(\mathcal{D},\theta) = \sum_{(x,y,\mathcal{C}) \in \mathcal{D}} \left[\phi(x,y,\mathcal{C}) - \sum_{y' \in \mathcal{C}} p(y'|x,\mathcal{C},\theta) \phi(x,y',\mathcal{C}) \right] \tag{2.2}$$

Log-Linear Models of Spatial Relations Above is a general description of log-linear models that can be used in a wide range of a applications. In order to use a log-linear model to score a triple consisting of a target object, reference object, and relation in a given context, we must decide how to partition the members of this triple into the input x and output y variables of the log-linear model.

We consider two different instantiations of the log-linear model for scoring spatial relations: one for generating spatial descriptions and one for interpreting them. A spatial description refers to a target object o using an utterance w that consists of two parts:

- A spatial relation $w.r$ (e.g., *right of*) from a fixed set of possible relations.
- A reference object $w.o$ (e.g., O_2) from the set of objects in the context.

When training a speaker model – one that generates a spatial description of a target object o – we treat o as the input in the log-linear model and the utterance w as the output y:

$$p(w|o, \mathcal{C}, \theta_{\text{S}})$$

Conversely, when training the listener, we take the input to be the utterance, and use the model to assign scores to the target object o:

$$p(o|w, \mathcal{C}, \theta_{\text{L}})$$

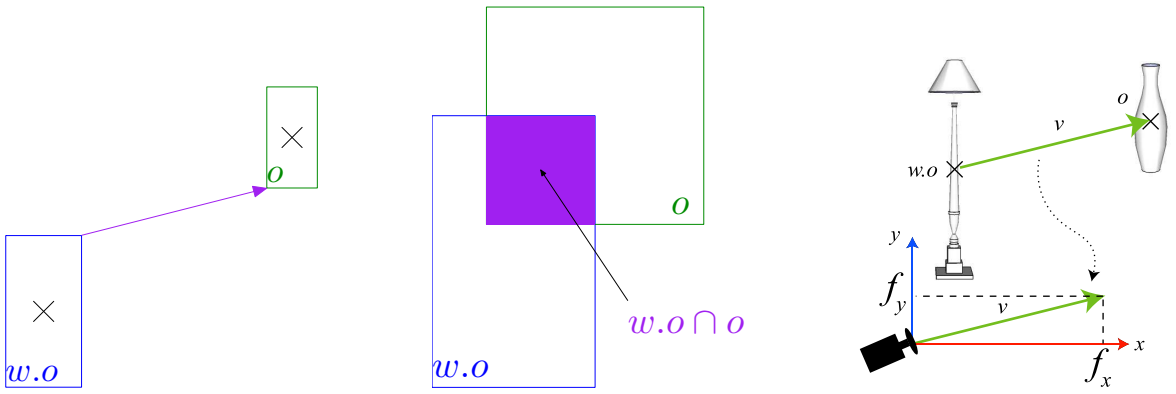
Here θ_{S} and θ_{L} are the parameter vectors for speaker and listener respectively.

We use the same set of features for both the speaker and listener models, but these models differ in the particular values of the parameters. Furthermore, the first normalization sums over possible utterances w while the second normalization sums over possible target objects o in the context $\mathcal{C}$. We learn the parameter vectors by optimizing the log-likelihood of the annotated training data under the respective models (Section 2.1).

2.3 Features

The crux of a log-linear model is the feature function. Our model aims to capture patterns in human judgments about binary spatial relations, and so our feature function must measure properties relevant to the meaning of the set of topological and projective prepositions under consideration. In our log-linear models, the features $\phi(o, w, \mathcal{C})$ are a function of the visual context $\mathcal{C}$, the target object o, and the spatial description w, which consists of the relation $w.r$ and the reference object $w.o$.

We are primarily concerned with modeling the meaning of spatial relations in terms of the object location, and therefore we approximate the representation of each object in the $3D$ scene by its bounding box: the smallest rectangular prism containing the object. The following are functions of the camera, target object o, and the reference object $w.o$ in the utterance. The full set of features is obtained by conjoining these functions with indicator



(a) Proximity Features *(b) Topological Features* *(c) Projection Features*

Figure 2.3: The features computed between the target (o) and reference (w.o) objects.

functions of the form $\mathbb{I}[w.r = r]$, where r ranges over the fixed set of prepositions under consideration.

- *Proximity functions* measure the distance between o and $w.o$. This is implemented as the minimum over all the pairwise Euclidean distances between the corners of the bounding boxes (see Figure 2.3a). We also have indicator functions for whether o is the closest object, among the top 5 closest objects, and among the top 10 closest objects to $w.o$.
- *Topological functions* measure containment between o and $w.o$: $vol(o \cap w.o)/vol(o)$ and $vol(o \cap w.o)/vol(w.o)$ (see Figure 2.3b). To simplify volume computation, we approximate each object by a bounding box that is aligned with the camera axes.
- *Projection functions* measure the relative position of the bounding boxes with respect to one another. Specifically, let v be the vector from the center of $w.o$ to the center of o. There is a function for the projection of v onto each of the axes defined by the camera orientation (see Figure 2.3c). Additionally, there is a set of indicator functions that capture the relative magnitude of these projections. For example, there is a indicator function denoting whether the projection of v onto the camera's x-axis is the largest of all three projections.

2.4 Extended Features

To adapt our models to the physical setting where the experiments involved scoring spatial relations between physical objects (Chapter 3 and Chapter 5), we extend the features above by including additional features and performing model selection. These extended features are inspired by Guadarrama and Pancho (2010), Regier and Carlson (2001), and Gorniak and Roy (2004).

The complete set of extended features include the following:

- *Simple Features* are functions only of the center of mass (CM) of the bounding boxes, and are comprised of: the Euclidean distance between the center of mass (CM) and the offsets in X, Y, Z between the CMs.
- *Complex Features* are functions of the relation between the bounding boxes (BBs) and are comprised of: the percentage of overlap of the BBs, the percentage that the target BB is inside the landmark BB, the minimum distance between the BB, and whether or not the target BB is in contact with the landmark BB.
- *Psycholinguistic Features* extend those presented in Regier and Carlson (2001) to $3D$ objects and to all projective prepositions.

Using these sets of features we defined the following models (see Fig. 2.4 for results):

- *Simple Model* uses simple features.
- *Complex Model* uses simple and complex features.
- *Psycholinguistic Model* uses simple and psycholinguistic features.
- *Combined Model* uses all the features.

We train each of these four spatial preposition models on the same virtual dataset as before. Prior to deploying the model learned on virtual environments to the physical domain, we perform model selection via a development set of 100 annotated examples of relations pertaining to physical objects arranged in a physical scene. Specifically, we define a *Hybrid Model* which, for each preposition, selects the best performing model from the four above. Empirically, we found that this method of model selection performed better than others.

2.4.1 Results of Extended Features

In this section we present the results of testing the performance of the extended features in the physical setting.[2]

Given a reference object and a spatial preposition, our model predicts a target object. We evaluate the average count of how often the predicted object matches human judgment for the same task. We collected a testing set of 300 (landmark, spatial prepositions)-pairs from a physical context, and asked 3 separate people to (i) select the set of valid targets and (ii) pick a "best answer" from among that set. The ground truth is defined by majority vote.

As can be seen in Fig. 2.4, the random baseline for selecting the best answer is 14%, while the inter-annotator accuracy ("humans") is 85%. Human agreement is below 100% due to the inherent difficulty in selecting the best answer for some ambiguous prepositions. Indeed, our experiments indicate that the intrinsic ambiguity of certain queries is a substantial source of errors.

[2]For the sake of flow, we leave the discussion of the results of the simpler set of features to Section 4.6.

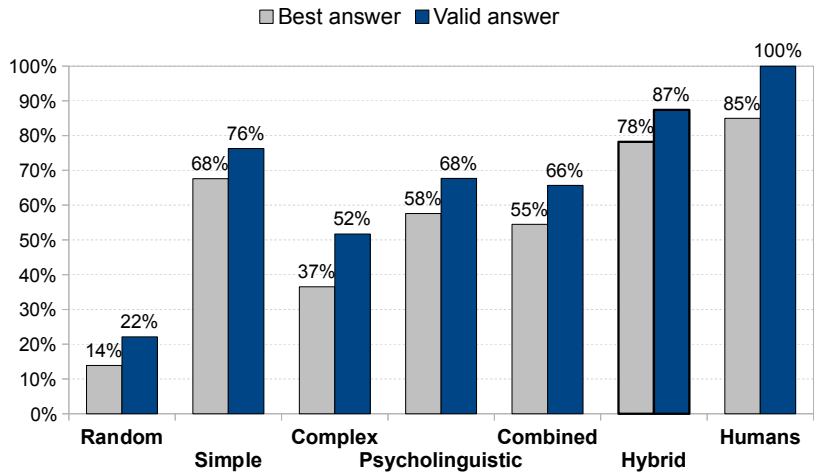


Figure 2.4: Spatial Prepositions Results

Because of the ability to use model selection to adapt to the physical setting, the best results are achieved with the Hybrid Model. We use the Hybrid Model as a component when building larger systems for interpretation (Chapter 3) and deploying on a robotic platform (Chapter 5).

2.5 Conclusion

The main contribution of this chapter is a computational model of preposition semantics. Rather than focusing on modeling the meaning of prepositions in isolation, we take a more pragmatic approach that explores how prepositions are used within language to refer to objects, or regions of space that objects can occupy. Our model captures the semantics of a preposition in so far as it is used to select a target object from amongst a set of distractor objects in a given context.

As we mentioned in Section 1.4.1, the precise meaning of a preposition is not clear cut and is often context dependent. The parameters of our log-linear model are, therefore, fit to model human judgements on data. Unlike some previous work, we use a single set of features to capture the meaning of both projective and topological prepositions. We collect data and train only on the most common set of binary argument English prepositions, thereby omitting many of the less common prepositions, prepositions that take more than two arguments (for example, *between*), and prepositions in other languages (which can function differently from English). Although we only collected data for a limited set of prepositions, we believe that our general log-linear approach to modeling the meaning of prepositions can be straightforwardly extended to cover a broader set by simply introducing new features or training on more data.

We chose this approach to modeling the semantics of prepositions because it is modular and lends itself well to be used in larger models. Indeed, the rest of this thesis explores how this model of spatial relations can be used in various applications such as: interpreting

spatial descriptions consisting of more than just a single relation (Chapter 3), generating spatial descriptions of objects (Chapter 4), and facilitating human robot interaction in a physical environment (Chapter 5).

Chapter 3

A Grounded Approach to Interpreting Spatial Descriptions

3.1 Introduction

In this chapter, we present a semantic parser that grounds the meaning of spatial descriptions in perceptual input from a computer vision system. Our parser takes as input a visual context such as the one presented in Figure 3.1 and a natural language utterance such as:

in the back behind the spam

The model interprets the utterance, and outputs the identity of the target object that is being referred to by the utterance (i.e. o_2).

To perform this interpretation, our model addresses several challenges. It recognizes the objects in the scene using a computer vision system (o_5 is `spam_can`). It determines the spatial relations that hold between these objects (o_2 is `behind` o_5, and o_2 is positioned on the `back`). It learns the alignment between natural language words and the concepts they represent (*spam* means `spam_can`). Finally, it composes the meanings of the individual concepts to build the meaning of the utterance as a whole.

We focus on the task of learning to predict the correct target object from natural language descriptions while treating the exact form of the semantic parse as unobserved. Our parser learns to ground semantic concepts in noisy perceptual data of real-world objects by integrating with computer vision systems. We train our parser over a syntactico-semantic

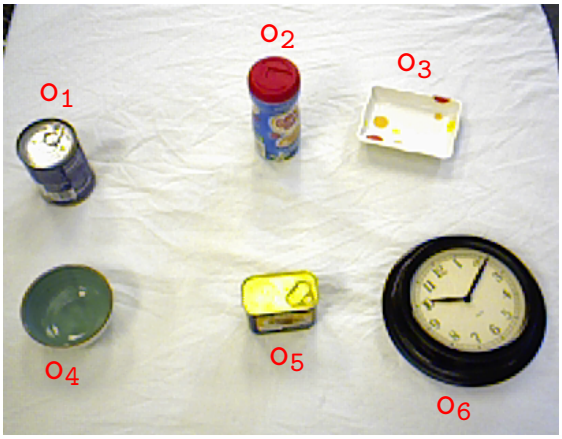


Figure 3.1: An example of the visual contexts used by our system to ground linguistic utterances.

grammar containing the fixed set of visual relations recognized by the computer vision systems. Although the set of semantic relations is fixed, the grammar with which they are expressed is fully broad-coverage and automatically fit. In general, our approach is that the language used should be unconstrained even if the grounded representations are not.

To learn a semantic parser from natural language utterances paired with the object to which they refer, we create a probabilistic model where the structure of the semantic parse appears as a latent variable. In contrast, most previous work in semantic parsing required annotations of logical forms during training (Zelle and Mooney, 1996; Tang and Mooney, 2001; Ge and Mooney, 2005; Zettlemoyer and Collins, 2005; Kate and Mooney, 2007; Zettlemoyer and Collins, 2007; Wong and Mooney, 2007; Kwiatkowksi et al., 2010).

A common approach to training semantic parsers from incomplete data is to anchor the grounding in a database containing facts that describe universally true relations in the external world (Poon and Domingos, 2009; Clarke et al., 2010; Liang et al., 2011). In contrast, we consider learning in the setting where there is no simple database of definitive relations that hold of objects in the world. Instead, our model learns from the more ambiguous perceptual judgments and confidence scores given by a computer vision system, where the judgments assigned to each relation vary with context (e.g. o_2 might be behind o_5 in one context, but not in another).

Other studies have explored training semantic parsers from more structured representations of the world such as: symbolic transcriptions of RoboCup sportscasts (Kim and Mooney, 2010; Chen et al., 2010; Börschinger et al., 2011); through navigation (Kollar et al., 2010; Matuszek et al., 2010; Vogel and Jurafsky, 2010); by following instructions (Branavan et al., 2009); by reading manuals (Branavan et al., 2011); or through conversations (Artzi and Zettlemoyer, 2011). Yet, all of these approaches rely on observations of the environment that contain no perceptual noise.

We jointly learn a linguistic grounding and semantic parser over the full space of semantic parses. In contrast, other work learns a grounding of individual words or short phrases

(Bailey et al., 1997; Barnard et al., 2003; Yu and Ballard, 2004), or learning a semantic alignment without semantic structure (Fleischman and Roy, 2007a; Gupta and Mooney, 2009). Some previous work builds a pipeline that performs semantic analysis over the output of a syntactic parser (Tellex et al., 2011b).

Our model trains a conditional random field with a latent, tree-structured semantic parse variable using a rich set of features to predict a target object from a natural language utterance. The features in the model capture the alignment between words and concepts as well as the object recognition and spatial relation judgments given by the computer vision system. Our model achieves 65.0% accuracy as compared to a direct association baseline of 46.5% and human performance of 86.5%. To support the representational adequacy of our semantic model, we show that an external syntactic bias does not improve performance. To support the effectiveness of our structured interpretations, we show that features that bypass the structured interpretation by predicting the output directly from the words also do not improve performance.

3.2 Overview

We began by collecting a new dataset using Amazon Mechanical Turk. For a given visual context $\mathcal{C}$ (such as the one shown in Figure 3.1) and target object $\texttt{O}$, we prompted annotators to generate natural language utterances x (e.g. *in the back behind the spam*) describing the target object spatially in relation to other objects in the context (details about data collection in Section 3.4). Using this data, we built a latent variable model to induce a semantic parse y: $p(\texttt{O}, y|x, \mathcal{C})$.

We treat y as a tree-structured latent variable coming from a manually specified semantic grammar defined in Section 3.3.2. The semantic parser is influenced by a computer vision system that processes the visual context and, in some experiments, a syntactic parser to analyze the input utterance.

We considered the effect of using a broad coverage, high-accuracy syntactic parser to analyze the input utterance and loosely inform the semantic parse of x. The syntactic parser takes the utterance x as input and outputs a ranked list of syntactic parses. We computed structural agreement features between the syntactic parses and the semantic parse, which lets the syntactic parser inform the semantic parser, while still allowing divergence when necessary.

The visual context is processed using a computer vision system consisting of an object model for nouns and two spatial relation models, one unary (for absolute location in the scene) and one binary (for relative locations between two objects). The overall structure of the model is shown in Figure 3.2.

The noun object model is an object recognition system responsible for assigning a real valued score $\textsc{noun}_{\mathcal{C}}(\texttt{O}, \texttt{N})$ to pairs of an object $\texttt{O}$ and a noun concept $\texttt{N}$, where larger values correspond to higher confidence that $\texttt{O}$ is an instance of $\texttt{N}$ with respect to $\mathcal{C}$. For example,

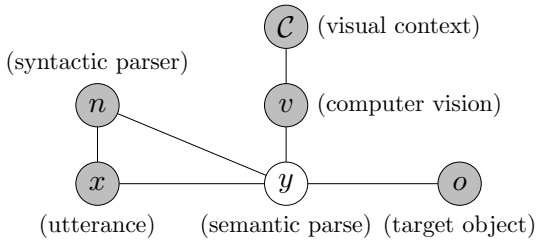


Figure 3.2: The structure of our model. A natural language utterance x in visual context $\mathcal{C}$ is made in reference to target object O. We induce a semantic parse y that maps the utterance to the target object. The semantic parse is informed by a computer vision system v that processes the visual context, and (in some experiments) syntactic parses n of x.

the noun model score $\textsc{noun}(\mathrm{O}_5, \texttt{spam\_can})$ for assigning the concept label `spam_can` to the object O_5 should be high in Figure 3.1.

The unary spatial relation model assigns a real valued score $\textsc{unary}_{\mathcal{C}}(\mathrm{O}, \mathtt{R})$ between object O and spatial relation $\mathtt{R}$, where larger values correspond to higher confidence that O satisfies $\mathtt{R}$ in $\mathcal{C}$. For example, given the configuration of objects in Figure 3.1, the model should assign a high score to $\textsc{unary}_{\mathcal{C}}(\mathrm{O}_5, \texttt{bottom})$.

The binary spatial relation model assigns a real valued score $\textsc{binary}_{\mathcal{C}}(\mathrm{O}, \mathtt{R}, \mathrm{O}')$ between pairs of objects O, O' and a relation $\mathtt{R}$, where larger values correspond to higher confidence that O stands in relation $\mathtt{R}$ to O' in $\mathcal{C}$. For example, given the configuration of objects in Figure 3.1, the model should assign a high score to $\textsc{binary}(\mathrm{O}_2, \texttt{behind}, \mathrm{O}_5)$.

All the base vision models are trained offline (see Section 3.3.1).

Although we have chosen to focus on noun recognition and spatial relations, our model is extensible, and can combine various sources of perceptual input in a probabilistically sound way. Our semantic grammar gracefully ignores concepts that fall outside the scope of the vision models, guaranteeing that the parser will return a semantic analysis for any utterance. The partially supervised nature of the model allows it to be trained from data which is easy to collect, reducing the burden on annotators.

Given our design, we sketch the limitations of the choices we made and indicate opportunities for future work. Our parser is only capable of assigning meaning using the concepts present in the base models, which omits concepts such as adjectives, dynamic prepositions, and verbs. Our noun model only contains concepts for individual objects, precluding reference to plurals or subsets of objects such as rows or columns. The parser is robust to linguistic content that falls outside the domains of the vision models, but when there is insufficient remaining content to interpret the utterance, our model will likely return an incorrect result. The parser does not handle complex linguistic phenomena such as negation, quantification, deixis of place, or presupposition failure.

3.3 Model

3.3.1 Base Vision Models

The system perceives the visual context through the base noun object model and the unary and binary spatial relations models. These models serve a similar role to the databases in previous work (Clarke et al., 2010; Liang et al., 2011) because they provide a set of relations describing the external world upon which the language is grounded. However, unlike databases, our base vision models do not provide relations that are simply true, but instead relations that hold with some degree of confidence. That is to say, these models provide many spurious relations which are distinguished from true relations only in that the spurious relations receive a lower confidence score. Furthermore, the vision models are empirically learned and so may even predict relations which are simply incorrect. Our model uses real valued outputs of the vision system as feature values. A method to ground meaning upon uncertain perception is one of the primary contributions of this work.

Noun Object Model

For a given visual context $\mathcal{C}$, the noun object model assigns a score to an object and one of a fixed set of 50 noun concepts $\texttt{N} \in \mathcal{N}$. For example, the noun model might assign a score to object $\texttt{O}$ and concept `spam_can` that corresponds to the confidence with which the noun model believes that object $\texttt{O}$ should be labeled with `spam_can` in the given context.

To recognize the objects in the image, we apply the state-of-the-art *instance* recognition algorithm presented in Jia et al. (2012). The instance recognizer takes as input image patches containing objects, extracts local color patches, forms a codebook, pools features in a grid, and then assigns a score using a linear SVM. These scores are normalized, to put them on a common scale, then squared and passed through the logarithm function. The SVM is trained offline on multiple images taken of 50 kitchen and office objects that were jittered and rotated to increase robustness to perceptual changes.

To apply the offline-trained object recognition model to our online setting, we segmented the input image by projecting the 3D bounding boxes (described below) onto the image plane and passed each image segment containing an object to the classifier. The algorithm correctly labels the objects with 93.8% accuracy.

Binary Spatial Relation Model

For a given context $\mathcal{C}$, the binary spatial relation model assigns a score to an ordered pair of an object and one of a fixed set of 10 binary spatial relation concepts $\texttt{R} \in \mathcal{R}_2$. For example, the binary spatial relation model assigns a score to objects $\texttt{O}, \texttt{O}'$ and the relation `behind` corresponding to the confidence with which the model believes that $\texttt{O}$ is behind $\texttt{O}'$.

To score a relation between objects, we extend the model presented in Section 2.4. The model takes as input the 3D bounding boxes of the two argument objects and computes a set of features of the bounding boxes. The features include the amount of overlap between the bounding boxes, the distance between the bounding box centers, and information related to

the angles formed between the vector connecting the bounding box centers and the camera axes. A logistic regression model of $p(\mathtt{O} \mid \mathtt{R}, \mathtt{O}', \mathcal{C})$ is trained offline on 3D Google Sketchup models, and the binary spatial relation model returns the logarithm of this probability.

To adapt the binary spatial relation model that was trained offline on 3D models to our online, real-world setting, we manually rectified the camera position and computed 3D bounding boxes for the objects in the scene. To extract the bounding boxes, we used a 3D Asus Xtion camera, subtracted points that fell on the plane of the background, and clustered the resulting point clouds. On its own, the segmentation achieves 94.0% accuracy, where most errors are caused by over-segmenting large objects. Spurious segments were manually discarded, resulting in some bounding boxes that are smaller than ideal. Nevertheless, the binary spatial relation model correctly identifies the target object for a given (relation, reference object) pair with 92.2% accuracy.

Unary Spatial Relation Model

For a given context $\mathcal{C}$, the unary spatial relation model assigns a score between an object and one of a fixed set of relations $\mathtt{R} \in \mathcal{R}_1$.[1] For example, the unary spatial relation model might assign a score between object $\mathtt{O}$ and the relation `back` corresponding to the confidence with which the model believes that object $\mathtt{O}$ is in the `back` in the given context.

To score a relation, we adapt the model presented in Section 2.4. We train a separate logistic regression model of $p(\mathtt{O} \mid \mathtt{R}, \mathtt{O}', \mathcal{C})$ using the same features on a new 3D Google Sketchup dataset, different from the one used to train the binary spatial relation model. Whereas in the binary model the reference object $\mathtt{O}'$ was a variable, in the unary model we hold the reference object $\mathtt{O}'$ fixed to be an invisible, central anchor object that remains unchanged in each visual context. The output of the unary spatial relation model is the logarithm of the probability predicted by the logistic regression.

To apply the unary spatial relation model trained offline to our online setting, we again rectify the camera position and use the same 3D bounding boxes computed for the binary spatial relations model. The bounding box of the anchor object was manually specified for each visual context. The unary spatial relation model labels each object with a correct relation with 100% accuracy.

3.3.2 Semantic Grammar

The semantic grammar defines a mapping between a natural language utterance, consisting of a sequence of words w, and the specific real world object $\mathtt{O}$ to which the utterance refers. This mapping is mediated by a closed class of noun concepts $\mathcal{N}$, unary spatial relation concepts $\mathcal{R}_1$, and binary spatial relation concepts $\mathcal{R}_2$ recognized by the computer vision system.

Our semantic grammar contains terminal rules for mapping words to concepts, unary rules for mapping noun and unary relation concepts to objects, and binary rules for mapping

[1] The unary spatial relations our model supports are: `left`, `right`, `back`, `bottom`, `middle`, `corner`.

Terminal Rules

for each word w, $\texttt{N} \in \mathcal{N}$, $\texttt{R}_1 \in \mathcal{R}_1$, $\texttt{R}_2 \in \mathcal{R}_2$:

$$\texttt{N} \rightarrow w$$

$$\texttt{R}_1 \rightarrow w$$

$$\texttt{R}_2 \rightarrow w$$

$$\texttt{NULL} \rightarrow w$$

Unary Rules

for each $\texttt{O} \in \mathcal{C}$, $\texttt{N} \in \mathcal{N}$, $\texttt{R} \in \mathcal{R}_1$:

$$\texttt{O} \rightarrow \texttt{N}$$

$$\texttt{O} \rightarrow \texttt{R}$$

Binary Rules

for each $\texttt{O}, \texttt{O}' \in \mathcal{C}$, $\texttt{N} \in \mathcal{N}$, $\texttt{R}_1 \in \mathcal{R}_1$, $\texttt{R}_2 \in \mathcal{R}_2$:

$$\texttt{O} \rightarrow \texttt{R}_2\ \texttt{O}' \quad | \quad \texttt{O}\ \texttt{O}$$

$$\texttt{N} \rightarrow \texttt{NULL}\ \texttt{N} \mid \texttt{N}\ \texttt{NULL}$$

$$\texttt{R}_1 \rightarrow \texttt{NULL}\ \texttt{R}_1 \mid \texttt{R}_1\ \texttt{NULL}$$

$$\texttt{R}_2 \rightarrow \texttt{NULL}\ \texttt{R}_2 \mid \texttt{R}_2\ \texttt{NULL}$$

Figure 3.3: The semantic grammar for a visual context $\mathcal{C}$ is built over the set of noun labels $\mathcal{N}$, unary spatial relations $\mathcal{R}_1$, and binary spatial relations $\mathcal{R}_2$ recognized by our computer vision system.

between two objects via a binary relation or conjoining two descriptions of the same object. Following Börschinger et al. (2011), our grammar includes a special concept NULL to absorb words that fall outside the concepts present in the computer vision system, and there are additional binary rules for consuming these NULL concepts. The exact form of our grammar for a visual context $\mathcal{C}$ is shown in Figure 3.3.

An example of a semantic parse is shown in Figure 3.4. Each parse in our grammar represents one possible interpretation of the utterance. The parse synthesizes information

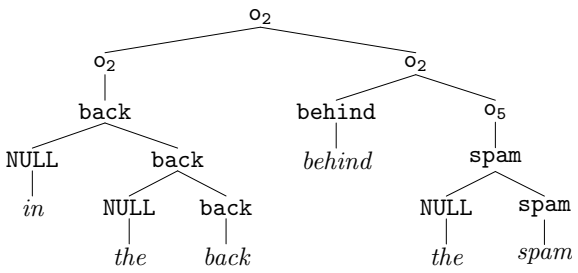


Figure 3.4: An example semantic parse with respect to Figure 3.1. Internal nodes are either a noun concept (spam_can), a unary relation concept (back), a binary relation concept (behind), null concept (NULL), or an object in the scene (o_2, o_5). The target object referred to by the utterance (o_2) is at the root.

from many data sources to predict a target object. It specifies an alignment between words and concepts, a labeling of a subset of objects in the visual context with noun concepts, a choice about which relations hold between objects, and the syntactic structure governing how these relations combine. The target object referred to by the semantic parse is found at the root. The example parse demonstrates how we use the NULL concept to consume words without impacting meaning, which allows for robust interpretation: our parser will return an interpretation for every sentence.

Börschinger et al. (2011) propose a similar syntactic-semantic grammar formalism for semantic parsing, however theirs is a PCFG where the score assigned to each parse is the product of the multinomial weight associated with each production rule. In contrast, our parses are scored via a conditional random field over trees (see Section 3.3.4), which allows us to include features that ground into the computer vision system (see Section 3.3.3).

3.3.3 Features

To enable efficient dynamic programming, the features in our model factor along the production rules in the parse tree. Our features are divided into the following categories.

Alignment Features For each concept $\mathtt{A} \in \mathcal{N} \cup \mathcal{R}_1 \cup \mathcal{R}_2$ and word w, there is a feature that counts the number of occurrences of production $\mathtt{A} \to w$ in the parse tree.

Base Vision Model Features For each object $\mathtt{O}$ and noun concept $\mathtt{N}$ that form a production rule $\mathtt{O} \to \mathtt{N}$ in the parse, the BaseNounModel feature is incremented with the score the base noun object model gives for assigning object $\mathtt{O}$ the label $\mathtt{N}$ in the given context: $\textsc{noun}_{\mathcal{C}}(\mathtt{O}, \mathtt{N})$.

For each object $\mathtt{O}$ and unary relation concept $\mathtt{R}$ that form a production rule $\mathtt{O} \to \mathtt{R}$ in the parse, the BaseUnaryModel feature is incremented with the score that the base

unary spatial relation model gives to labeling object $\mathtt{O}$ with relation $\mathtt{R}$ in the given context: $\textsc{unary}_{\mathcal{C}}(\mathtt{O}, \mathtt{R})$.

For every pair of objects $(\mathtt{O}, \mathtt{O}')$ and binary relation concept $\mathtt{R}$ that form a production rule $\mathtt{O} \rightarrow \mathtt{R}\ \mathtt{O}'$ in the parse, the BaseBinaryModel feature is incremented by the score the base binary relation model assigns to the relation $(\mathtt{O}, \mathtt{R}, \mathtt{O}')$ in $\mathcal{C}$: $\textsc{binary}_{\mathcal{C}}(\mathtt{O}, \mathtt{R}, \mathtt{O}')$.

Syntactic Parse Features In some experiments we consider the impact of using an external syntactic parser to influence the structure of the predicted parses. Specifically, we extract the top k binarized parses from the Berkeley Parser (Petrov and Klein, 2007) for the given utterance.[2] For each of the k parses, we have a separate set of features that measure agreement in structure between our predicted parse and the Berkeley parse. We have a feature that counts the number of constituents in agreement, a feature that counts the number of constituents that share a starting point, and a feature that counts the number of constituents that share an end point between the two parses.

Direct Association Features In some experiments we consider direct association features, which learn a direct mapping from every n-gram appearing in the utterance to the object at the root. Specifically, for a parse with root $\mathtt{O}$ in visual context $\mathcal{C}$, there is a separate feature that counts the number of times each n-gram appears in the utterance. In our experiments $n \in \{1, 2\}$.

3.3.4 Learning

During training, we observe triples of a visual context $\mathcal{C}$, utterance x, and gold target object $\mathtt{O}$. The semantic parse y is a tree-structured latent variable with leaves x and root $\mathtt{O}$. Using these triples and the features in Section 3.3.3, we define a latent variable conditional random field (Sutton and McCallum, 2007) and train it to induce the semantic parse.

We score each parse $y \in Y(x)$ using a log-linear model:

$$p_\theta(y \mid x, \mathcal{C}) = \frac{\exp \theta^T F(x, y, \mathcal{C})}{\sum_{y'} \exp \theta^T F(x, y', \mathcal{C})}$$

where θ is the vector of parameters.[3]

Given a dataset of triples $\{x_i, \mathtt{O}_i, \mathcal{C}_i\}_{i=1}^n$ where each utterance x describes a target object $\mathtt{O}$ in visual context $\mathcal{C}$, we use L-BFGS to optimize the regularized marginal log-likelihood of our training set:

$$L(\theta) = \sum_i \log \sum_{y \in T(\mathtt{O}_i)} p_\theta(y \mid x_i, \mathcal{C}_i) - \lambda \|\theta\|_2^2$$

where $T(\mathtt{O}) = \{y \mid \textsc{Root}(y) = \mathtt{O}\}$ is the set of parses whose root symbol is $\mathtt{O}$.

[2] In our experiments, $k = 5$.

[3] To prime the model to believe the vision systems, the parameters for all base vision model features are initialized with positive weights.

The gradient of our objective is:

$$\nabla L(\theta) = \sum_i \mathbb{E}_\theta \left[F(x_i, y, \mathcal{C}_i) \mid y \in T(\mathrm{o}_i) \right] - \mathbb{E}_\theta \left[F(x_i, y, \mathcal{C}_i) \right] - 2\lambda\theta$$

The gradient requires computing a difference of two expectations. The first is the expectation over all semantic parses such that the interpretation of the semantic parse matches the observed target object. The second is the expectation over all semantic parses.

We use the Inside-Outside algorithm to efficiently compute both expectations. When summing over all parses, we initialize the outside pass equally for all objects $\mathrm{o} \in \mathcal{C}$, and to sum only over the parses in $T(\mathrm{o})$, we initialize the outside pass to zero everywhere except o.

3.4 Data

We use Amazon Mechanical Turk (MTurk) to collect data. In our MTurk task, users are shown an image of a tabletop containing a set of objects where a single object is highlighted. They are prompted to describe the location of the highlighted object with respect to the other objects. To discourage ambiguous descriptions, we prompted users to provide a description that would suffice for another person to guess the referenced object.

We had 8 different scenes, with an average of 6.125 objects per scene. We minimally process the collected descriptions by discarding empty submissions, automatically correcting spelling, converting to lowercase, and removing identical descriptions. After processing, 925 (scene, target object, description) triples remained, which we randomly split into train (624), test (200), and development (101) datasets. The utterance lengths ranged between 1 and 39 words with an average length of 12 words.[4]

The utterances we collected are complex, containing a wealth of linguistic phenomena not captured by the semantic representation of our models. Because the utterances describe a visual scene composed of real physical objects, there are many subtle visual cues from which the annotators can draw inspiration for their descriptions. For example, consider some of the various ways annotators referred to the spam in Figure 3.1:

- *can of spam*
- *canned meat*
- *pop-top spam can*
- *rectangular can*
- *tin food*
- *blue can with the pull tab opening*

These descriptions refer to shape, material, color, and function; all of which fall outside the scope of our base vision systems.

[4]The collected data and output of the vision models will be published at `http://`*ANONYMIZED*.

Rather than discarding data that does not fit our model, or pursuing the endless task of building a model complex enough to capture all the nuances in the data, we decide to structure our model so that interpretable phenomena are directly captured, while non-interpretable phenomena are robustly absorbed as noise.

3.5 Experiments

We compare the accuracy of our semantic parser to two baselines and a human skyline. The results of these experiments are presented in Table 3.1.

Our semantic parser is trained using the alignment and base vision model features. The random baseline guesses the target object uniformly at random from the set of objects in the visual context.

Our second baseline is the direct association model, which learns a logistic-regression model over the direct association features defined in Section 3.3.3 to predict the target object o directly from the unigrams and bigrams in the utterance x, bypassing the semantic parse: $p(\mathrm{o} \mid x, \mathcal{C})$.

To explore the adequacy of our semantic representation for the range of semantic phenomena covered by our model, in Section 3.5.3 we report the effects of adding the syntactic and direct association features to our model.

3.5.1 Evaluation

Given a model $\mathcal{M} = p(\mathrm{o}|x, \mathcal{C})$ and a test set $\mathcal{D} = \{(x_i, \mathrm{o}_i, \mathcal{C}_i)\}_{i=1}^{n}$, where each test instance consists of a natural language utterance x made in reference to a target object o in visual context $\mathcal{C}$, we define the accuracy of the model on the test set:

$$\textsc{Acc}(\mathcal{M},\mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{(x,\mathrm{o},\mathcal{C})\in\mathcal{D}} \mathbb{I}[\mathrm{o} = \arg\max_{\mathrm{o}'} p(\mathrm{o}'|x, \mathcal{C})]$$

as the average number of times the model correctly predicts which target object the utterance is refers to.

3.5.2 Results

The direct association baseline directly predicts the target object from the unigrams and bigrams in the utterance. By directly predicting the target object from the words,

Model	Accuracy
Random baseline	16.3%
Direct association	46.5%
Our model	**65.0%**
Human annotators	86.5%

Table 3.1: Comparison of our model against two baselines and a human skyline.

the direct association baseline bypasses the semantic grammar and the base vision models, and is therefore is not subject to their limitations. However, the direct association baseline performs substantially worse than our model (46.5% vs 65.0%), demonstrating the benefit of constructing a deep semantic analysis of the utterance. The direct association baseline is unable to leverage information learned about the target objects in one visual context to the disjoint set of target objects in another visual context. In other words, if the direct association baseline has not seen an object during training, it is incapable of predicting the object during test time.

In contrast, our model uses semantic concepts to decouple the words from the scene-specific target object. The alignment features capture the alignment between words and scene-independent concepts, thereby allowing our model to generalize meaning across visual contexts. To ground the set of concepts in a particular context, our model relies on the computer vision system through the base vision model features.

Most errors can be attributed to the limiting assumptions of our semantic grammar. As mentioned in Section 3.4, the utterances often reference concepts outside of the domain of the base vision models. For example, the following description refers to object o_2 in Figure 3.1: *at the top of a triangle formed by the clock and the bowl*. To interpret this utterance correctly, the model would need to interpret the word *triangle*, infer that an equilateral triangle is most salient in this context, and be able to determine which object would complete a triangle formed by the other two objects. Another source of errors is the discrepancy between train and test vocabulary. For example, the word *pepsi* was used to refer to a coke can, and although the words *coke*, *cola*, *soda*, and *can* were all seen in training, *pepsi* was not.

To put these results in perspective, we report human accuracy. For each utterance x in our test set, we asked three separate annotators on MTurk to guess which object is referred to by x. These annotators guessed the correct target object with 86.5% accuracy. Incorrect predictions can be attributed to a number of factors, including: true but ambiguous reference descriptions (e.g. *in the same row as the spam*, which can refer to two objects); incorrect object reference (e.g. referring to a soda can as a *ball*); confusion in the meaning of *left* and *right*; and disagreement in perspective about which object is in front and which is behind.

3.5.3 Adequacy of Representation

To demonstrate the adequacy of our representation for the range of semantic phenomena covered by our models, we explore the impact of enabling additional sets of features. When we enable the syntactic features in addition to the alignment features and the base vision model features, we achieve 64.5% accuracy.

The motivation for including syntactic features is to resolve syntactic ambiguities that affect the interpretation. Our semantic grammar is syntactically and semantically ambiguous: the meaning of two adjacent prepositional phrases is ambiguous between whether the two phrases are conjoined and therefore describing the same object, or whether the second phrase is subordinate, modifying the noun in the first prepositional phrase. However, current syntactic parsers are still not very good at prepositional phrase attachment, particularly when tested out of their training domain. These ambiguities are perhaps better resolved by a semantic model. Given the weaknesses of the parser and the challenges of testing in a transductive setting, we find that adding an off-the-shelf, external model of syntax does not provide additional power to our model.

When we additionally enable the direct association features, we achieve 64.5% accuracy. This result suggests that conditioned on the concepts, the words provide little additional information for determining the target object.

3.6 Conclusion

The primary goal of this system is to learn to interpret broad syntactic realizations of references to objects by grounding them into a closed class of perceptual concepts. Despite the latent variable limitations and restricted concept space, this model still performs at 65.0%, which is halfway between a trained baseline and human performance.

It is noteworthy that although the model can only represent a small set of concepts, it is still capable of generating an interpretation for any sentence. The model's grammar learns from data what to interpret and what to ignore, and thereby projects each sentence into the restricted space of interpretable concepts. One avenue for expanding on this work is to broaden the coverage of the model to include additional concepts. The errors on the test data suggest that the most fruitful concepts to model are spatial semantic concepts pertaining to the arrangement of objects into groups such as rows and columns. Additionally, future work could address the issue of data sparsity that arises from lexical variation by exploiting information about which distinct phrases refer to the same underlying concept. For example, a model that was aware how words are grouped into synonyms would not need to separately learn that *soda* and *pop* refer to the same concept.

Aside from broadening the semantic concepts of the model, there are several ways in which the pragmatic coverage can be extended. The current model only can interpret references to objects; however, as mentioned in Section 1.2, there is more to language that just reference

— there is a rich variety of speech acts that are not covered by our model.[5] Furthermore, there are two types of sentences which should not be interpreted by the system. First, it is impractical to build a set of concepts that are all-encompassing and therefore some sentences will fall outside the scope of concepts included in the model, even if the set of concepts is broadened. The system should be able to detect when a sentence is outside of domain and fail rather than attempting to interpret the sentence. Second, some sentences will simply fail to refer to any object in the context. Namely, if the sentence has a presupposition failure, there will be an implicit false assumption about the context. Rather than attempting to interpret sentences with presupposition failure (the current behavior), it should be able to detect the failure and abort interpretation or ask for clarification.

[5]Chapter 5 does explore how a model that can interpret spatial descriptions can be extended to respond to questions and simple commands.

Chapter 4

A Game-Theoretic Approach to Generating Spatial Descriptions

4.1 Introduction

Language is about successful communication between a speaker and a listener. For example, if the goal is to reference the target object `O1` in Figure 4.1, a speaker might choose one of the following two utterances:

(*a*) *right of* `O2` (*b*) *on* `O3`

Although both utterances are semantically correct, (a) is ambiguous between `O1` and `O3`, whereas (b) unambiguously identifies `O1` as the target object, and should therefore be preferred over (a). In this chapter, we present a game-theoretic model that captures this communication-oriented aspect of language interpretation and generation.

Successful communication can be broken down into semantics and pragmatics. Most computational work on interpreting language focuses on compositional semantics (Zettlemoyer and Collins, 2005; Wong and Mooney, 2007; Piantadosi et al., 2008), which is concerned with verifying the truth of a sentence. However, what is missing from this truth-oriented view is the pragmatic aspect of language—that language is used to accomplish an end goal, as exemplified by speech acts (Austin, 1962). Indeed, although both utterances (a) and (b) are semantically valid, only (b) is pragmatically felicitous: (a) is ambiguous and therefore violates the Gricean maxim of manner (Grice, 1975). To capture this maxim, we develop a model of pragmatics based on game theory, in the spirit of Jäger (2008) but extended to the stochastic setting. We show that Gricean maxims fall out naturally as consequences of the model.

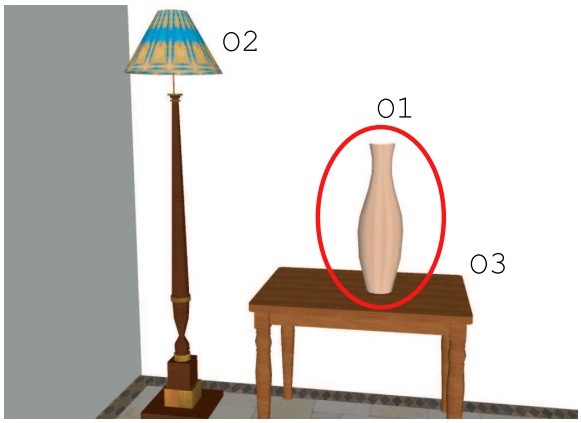


Figure 4.1: An example of a 3D model of a room. The speaker*'s goal is to reference the target object* `O1` *by describing its spatial relationship to other object(s). The* listener*'s goal is to guess the object given the speaker's description.*

An effective way to empirically explore the pragmatic aspects of language is to work in the grounded setting, where the basic idea is to map language to some representation of the non-linguistic world (Yu and Ballard, 2004; Feldman and Narayanan, 2004; Fleischman and Roy, 2007b; Chen and Mooney, 2008; Frank et al., 2009; Liang et al., 2009). Along similar lines, past work has also focused on interpreting natural language instructions (Branavan et al., 2009; Eisenstein et al., 2009; Kollar et al., 2010), which takes into account the goal of the communication. This work differs from ours in that it does not clarify the formal relationship between pragmatics and the interpretation task. Pragmatics has also been studied in the context of dialog systems. For instance, DeVault and Stone (2007) present a model of collaborative language between multiple agents that takes into account contextual ambiguities.

We present our pragmatic model in a grounded setting where a speaker must describe a target object to a listener via spatial description (such as in the example given above). Though we use some of the techniques from work on the semantics of spatial descriptions (Regier and Carlson, 2001; Gorniak and Roy, 2004; Tellex and Roy, 2009), we empirically demonstrate that having a model of pragmatics enables more successful communication.

4.2 Language as a Game

To model Grice's cooperative principle (Grice, 1975), we formulate the interaction between a speaker S and a listener L as a cooperative game, that is, one in which S and L share the same utility function. For simplicity, we focus on the production and interpretation of

single utterances, where the speaker and listener have access to a shared context. To simplify notation, we suppress writing the dependence on the context.

The Communication Game

1. In order to communicate a *target* o to L, S produces an *utterance* w chosen according to a strategy $p_S(w \mid o)$.
2. L interprets w and responds with a *guess* g according to a strategy $p_L(g \mid w)$.
3. S and L collectively get a utility of $U(o, g)$.

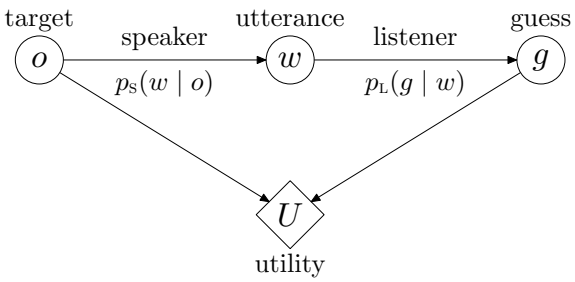


Figure 4.2: Diagram representing the communication game. A target, o, is given to the speaker that generates an utterance w. Based on this utterance, the listener generates a guess g. If $o = g$, then both the listener and speaker get a utility of 1*, otherwise they get a utility of* 0*.*

This communication game is described graphically in Figure 4.2. Figure 4.3 shows several instances of the communication game being played for the scenario in Figure 4.1.

Grice's maxim of manner encourages utterances to be unambiguous, which motivates the following utility, which we call *(communicative) success*:

$$U(o, g) \stackrel{\text{def}}{=} \mathbb{I}[o = g], \tag{4.1}$$

where the indicator function $\mathbb{I}[o = g]$ is 1 if $o = g$ and 0 otherwise. Hence, a utility-maximizing speaker will attempt to produce unambiguous utterances because they increase the probability that the listener will correctly guess the target.

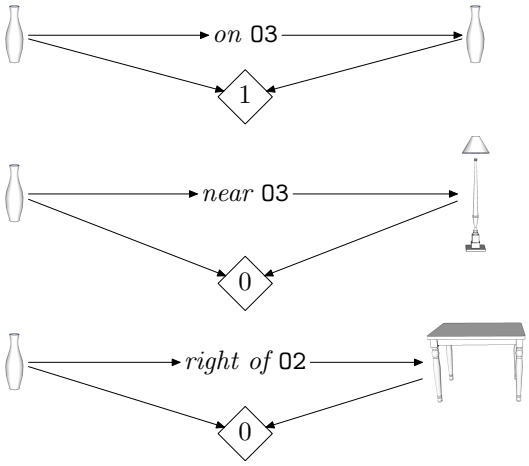


Figure 4.3: Three instances of the communication game on the scenario in Figure 4.1. For each instance, the target o, utterance w, guess g, and the resulting utility U are shown in their respective positions. A utility of 1 is awarded only when the guess matches the target.

Given a speaker strategy $p_{\text{S}}(w \mid o)$, a listener strategy $p_{\text{L}}(g \mid w)$, and a prior distribution over targets $p(o)$, the expected utility obtained by S and L is as follows:

$$\begin{aligned} \text{EU}(\text{S}, \text{L}) &= \sum_{o,w,g} p(o) p_{\text{S}}(w|o) p_{\text{L}}(g|w) U(o,g) \\ &= \sum_{o,w} p(o) p_{\text{S}}(w|o) p_{\text{L}}(o|w). \end{aligned} \tag{4.2}$$

4.3 From Reflex Speaker to Rational Speaker

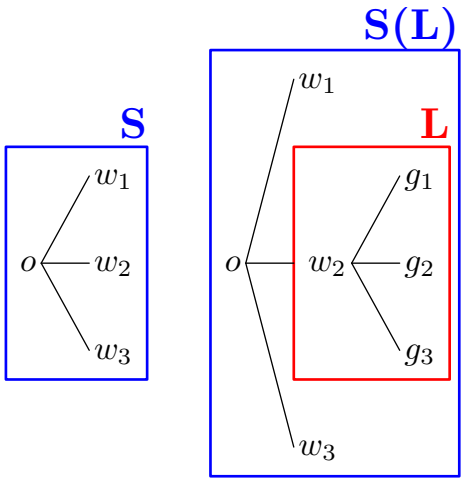
Having formalized the language game, we now explore various speaker and listener strategies. First, let us consider *literal* strategies. A literal speaker (denoted S:LITERAL) chooses uniformly from the set of utterances consistent with a target object, i.e., the ones which are semantically valid;[1] a literal listener (denoted L:LITERAL) guesses an object consistent with the utterance uniformly at random.

In the running example (Figure 4.1), where the target object is O_1, there are two semantically valid utterances:

$$(a)\ \textit{right of}\ \text{O}_2 \qquad (b)\ \textit{on}\ \text{O}_3$$

S:LITERAL selects (a) or (b) each with probability $\frac{1}{2}$. If S:LITERAL chooses (a), L:LITERAL will guess the target object O_1 correctly with probability $\frac{1}{2}$; if S:LITERAL chooses (b),

[1]Semantic validity is approximated by a set of heuristic rules (e.g. *left* is all positions with smaller x-coordinates).



(a) Reflex speaker (b) Rational speaker

Figure 4.4: (a) A reflex speaker (S) directly selects an utterance based only on the target object. Each edge represents a different choice of utterance. (b) A rational speaker (S(L)) selects an utterance based on an embedded model of the listener (L). Each edge in the first layer represents a different choice the speaker can make, and each edge in the second layer represents a response of the listener.

L:Literal will guess correctly with probability 1. Therefore, the expected utility $\text{EU}(\text{S:Literal}, \text{L:Literal}) = \frac{3}{4}$.

We say S:Literal is an example of a *reflex* speaker because it chooses an utterance without taking the listener into account. A general reflex speaker is depicted in Figure 4.4(a), where each edge represents a potential utterance.

Suppose we now have a model of some listener L. Motivated by game theory, we would optimize the expected utility (4.2) given $p_{\text{L}}(g \mid w)$. We call the resulting speaker S(L) the *rational* speaker with respect to listener L. Solving for this strategy yields:

$$p_{\text{S(L)}}(w \mid o) = \mathbb{I}[w = w^*], \text{ where}$$
$$w^* = \operatorname*{argmax}_{w'} p_{\text{L}}(o \mid w'). \tag{4.3}$$

Intuitively, S(L) chooses an utterance, w^*, such that, if listener L were to interpret w^*, the probability of L guessing the target would be maximized.[2] The rational speaker is depicted in Figure 4.4(b), where, as before, each edge at the first level represents a possible choice for the speaker, but there is now a second layer representing the response of the listener.

To see how an embedded model of the listener improves communication, again consider our running example in Figure 4.1. A speaker can describe the target object o_1 using

[2]If there are ties, any distribution over the utterances having the same utility is optimal.

either w_1 = *on* O_3 or w_2 = *right of* O_2. Suppose the embedded listener is L:LITERAL, which chooses uniformly from the set of objects consistent with the given utterance. In this scenario, $p_{\text{L:LITERAL}}(\text{O}_1 \mid w_1) = 1$ because w_1 unambiguously describes the target object, but $p_{\text{L:LITERAL}}(\text{O}_1 \mid w_2) = \frac{1}{2}$. The rational speaker S(L:LITERAL) would therefore choose w_1, achieving a utility of 1, which is an improvement over the reflex speaker S:LITERAL's utility of $\frac{3}{4}$.

4.4 From Literal Speaker to Learned Speaker

In the previous section, we showed that a literal strategy, one that considers only semantically valid choices, can be used to directly construct a reflex speaker S:LITERAL or an embedded listener in a rational speaker S(L:LITERAL). This section focuses on an orthogonal direction: improving literal strategies with learning. Specifically, we construct learned strategies from log-linear models trained on human annotations. These learned strategies can then be used to construct reflex and rational speaker variants—S:LEARNED and S(L:LEARNED), respectively.

4.4.1 Training a Log-Linear Speaker/Listener

We train the speaker, S:LEARNED, (similarly, listener, L:LEARNED) on training examples which comprise the utterances produced by the human annotators. Each example consists of a 3D model of a room in a house that specifies the $3D$ positions of each object and the coordinates of a $3D$ camera. When training the speaker, each example is a pair (o, w), where o is the input target object and w is the output utterance. When training the listener, each example is (w, g), where w is the input utterance and g is the output guessed object.

An utterance w consists of two parts:

- A spatial preposition $w.r$ (e.g., *right of*) from a set of possible prepositions.[3]
- A reference object $w.o$ (e.g., O_3) from the set of objects in the room.

We consider more complex utterances in Section 4.5.

Both S:LEARNED and L:LEARNED are parametrized by log-linear models:

$$p_{\text{S:LEARNED}}(w|o;\theta_{\text{S}}) \propto \exp\{\theta_{\text{S}}^{\top}\phi(o,w)\} \tag{4.4}$$

$$p_{\text{L:LEARNED}}(g|w;\theta_{\text{L}}) \propto \exp\{\theta_{\text{L}}^{\top}\phi(g,w)\} \tag{4.5}$$

where $\phi(\cdot,\cdot)$ is the feature vector (described in Section 2.3), θ_{S} and θ_{L} are the parameter vectors for speaker and listener. Note that the speaker and listener use the same set of

[3] We chose 10 prepositions commonly used by people to describe objects in a preliminary data gathering experiment. This list includes multi-word units, which function equivalently to prepositions, such as *left of*.

features, but they have different parameters. Furthermore, the first normalization sums over possible utterances w while the second normalization sums over possible objects g in the scene. The two parameter vectors are trained to optimize the log-likelihood of the training data under the respective models.

4.5 Handling Complex Utterances

So far, we have only considered speakers and listeners that deal with utterances consisting of one preposition and one reference object. We now extend these strategies to handle more complex utterances. Specifically, we consider utterances that conform to the following grammar:[4]

[noun]	N	$\rightarrow$	*something* $\mid$ O_1 $\mid$ O_2 $\mid$ $\cdots$
[relation]	R	$\rightarrow$	*in front of* $\mid$ *on* $\mid$ $\cdots$
[conjunction]	NP	$\rightarrow$	N RP^*
[relativization]	RP	$\rightarrow$	R NP

This grammar captures two phenomena of language use, conjunction and relativization.

- Conjunction is useful when one spatial relation is insufficient to disambiguate the target object. For example, in Figure 4.1, *right of* O_2 could refer to the vase or the table, but using the conjunction *right of* O_2 *and on* O_3 narrows down the target object to just the vase.
- The main purpose of relativization is to refer to objects without a precise nominal descriptor. With complex utterances, it is possible to chain relative prepositional phrases, for example, using *on something right of* O_2 to refer to the vase.

Given an utterance w, we define its *complexity* $|w|$ as the number of applications of the relativization rule, RP $\rightarrow$ R NP, used to produce w. We had only considered utterances of complexity 1 in previous sections.

4.5.1 Example Utterances

To illustrate the types of utterances available under the grammar, again consider the scene in Figure 4.1.

Utterances of complexity 2 can be generated either using the relativization rule exclusively, or both the conjunction and relativization rules. The relativization rule can be used to generate the following utterances:

- *on something that is right of* O_2

[4]Naturally, we disallow direct reference to the target object.

- *right of something that is left of* O_3

Applying the conjunction rule leads to the following utterances:

- *right of* O_2 *and on* O_3
- *right of* O_2 *and under* O_1
- *left of* O_1 *and left of* O_3

Note that we inserted the words *that is* after each N and the word *and* between every adjacent pair of RPs generated via the conjunction rule. This is to help a human listener interpret an utterance.

4.5.2 Extending the Rational Speaker

Suppose we have a rational speaker S(L) defined in terms of an embedded listener L which operates over utterances of complexity 1. We first extend L to interpret arbitrary utterances of our grammar. The rational speaker (defined in (4.2)) automatically inherits this extension.

Compositional semantics allows us to define the interpretation of complex utterances in terms of simpler ones. Specifically, each node in the parse tree has a *denotation*, which is computed recursively in terms of the node's children via a set of simple rules. Usually, denotations are represented as lambda-calculus functions, but for us, they will be distributions over objects in the scene. As a base case for interpreting utterances of complexity 1, we can use either L:LITERAL or L:LEARNED (defined in Sections 4.3 and 4.4).

Given a subtree w rooted at $u \in \{\text{N}, \text{NP}, \text{RP}\}$, we define the denotation of w, $[\![w]\!]$, to be a distribution over the objects in the scene in which the utterance was generated. The listener strategy $p_{\text{L}}(g|w) = [\![w]\!]$ is recursively as follows:

- If w is rooted at N with a single child x, then $[\![w]\!]$ is the uniform distribution over $\mathcal{N}(x)$, the set of objects consistent with the word x.
- If w is rooted at NP, we recursively compute the distributions over objects g for each child tree, multiply the probabilities, and renormalize (Hinton, 1999).
- If w is rooted at RP with relation r, we recursively compute the distribution over objects g' for the child NP tree. We then appeal to the base case to produce a distribution over objects g which are related to g' via relation r.

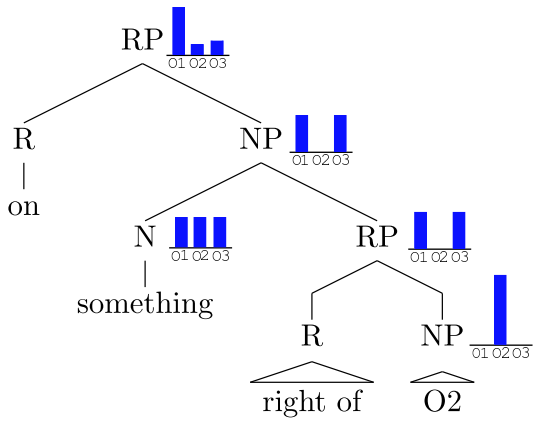


Figure 4.5: The listener model maps an utterance to a distribution over objects in the room. Each internal NP *or* RP *node is a distribution over objects in the room.*

This strategy is defined formally as follows:

$$p_{\text{L}}(g \mid w) \propto \begin{cases} \mathbb{I}[g \in \mathcal{N}(x)] & w = (\text{N}\ x) \\ \prod_{j=1}^{k} p_{\text{L}}(g \mid w_j) & w = (\text{NP}\ w_1 \ldots w_k) \\ \sum_{g'} p_{\text{L}}(g \mid (r, g')) p_{\text{L}}(g' \mid w') & w = (\text{RP}\ (\text{R}\ r)\ w') \end{cases} \tag{4.6}$$

Figure 4.5 shows an example of this bottom-up denotation computation for the utterance *on something right of* O_2 with respect to the scene in Figure 4.1. The denotation starts with the lowest NP node $[\![\text{O}_2]\!]$, which places all the mass on O_2 in the scene. Moving up the tree, we compute the denotation of the RP, $[\![\textit{right of}\ \text{O}_2]\!]$, using the RP case of (4.6), which results in a distribution that places equal mass on O_1 and O_3.[5] The denotation of the N node $[\![\textit{something}]\!]$ is a flat distribution over all the objects in the scene. Continuing up the tree, the denotation of the NP is computed by taking a product of the object distributions, and turns out to be exactly the same split distribution as its RP child. Finally, the denotation at the root is computed by applying the base case to *on* and the resulting distribution from the previous step.

Generation So far, we have defined the listener strategy $p_{\text{L}}(g \mid w)$. Given target o, the rational speaker S(L) with respect to this listener needs to compute $\text{argmax}_w\, p_{\text{L}}(o \mid w)$ as dictated by (4.3). This maximization is performed by enumerating all utterances of bounded complexity.

[5]It is worth mentioning that this split distribution between O_1 and O_3 represents the ambiguity mentioned in Section 4.3 when discussing the shortcomings of S:LITERAL.

4.5.3 Modeling Listener Confusion

One shortcoming of the previous approach for extending a listener is that it falsely assumes that a listener can reliably interpret a simple utterance just as well as it can a complex utterance.

We now describe a more realistic speaker which is robust to listener confusion. Let $\alpha \in [0, 1]$ be a *focus parameter* which determines the confusion level. Suppose we have a listener L. When presented with an utterance w, for each application of the relativization rule, we have a $1 - \alpha$ probability of losing focus. If we stay focused for the entire utterance (with probability $\alpha^{|w|}$), then we interpret the utterance according to p_{L}. Otherwise (with probability $1 - \alpha^{|w|}$), we guess an object at random according to $p_{\text{rnd}}(g \mid w)$. We then use (4.3) to define the rational speaker S(L) with respect the following "confused listener" strategy:

$$\tilde{p}_{\text{L}}(g \mid w) = \alpha^{|w|} p_{\text{L}}(g \mid w) + (1 - \alpha^{|w|}) p_{\text{rnd}}(g \mid w). \tag{4.7}$$

As $\alpha \to 0$, the confused listener is more likely to make a random guess, and thus there is a stronger penalty against using more complex utterances. As $\alpha \to 1$, the confused listener converges to p_{L} and the penalty for using complex utterances vanishes.

4.5.4 The Taboo Setting

Notice that the rational speaker as defined so far does not make full use of our grammar. Specifically, the rational speaker will never use the "wildcard" noun *something* nor the relativization rule in the grammar because an NP headed by the wildcard *something* can always be replaced by the object ID to obtain a higher utility. For instance, in Figure 4.5, the NP spanning *something right of* O_2 can be replaced by O_3.

However, it is not realistic to assume that all objects can be referenced directly. To simulate scenarios where some objects cannot be referenced directly (and to fully exercise our grammar), we introduce the *taboo setting*. In this setting, we remove from the lexicon some fraction of the object IDs which are closest to the target object. Since the tabooed objects cannot be referenced directly, a speaker must resort to use of the wildcard *something* and relativization.

For example, in Figure 4.6, we enable tabooing around the target O_1. This prevents the speaker from referring directly to O_3, so the speaker is forced to describe O_3 via the relativization rule, for example, producing *something right of* O_2.

4.6 Experiments

We now present our empirical results, showing that rational speakers, who have embedded models of listeners, can communicate more successfully than reflex speakers, who do not.

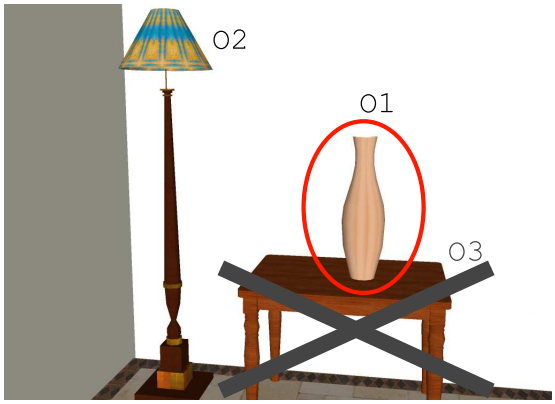


Figure 4.6: With tabooing enabled around o_1, o_3 can no longer be referred to directly (represented by an X*).*

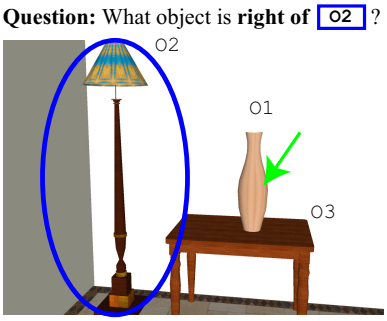


Figure 4.7: Mechanical Turk listener task: a human listener is prompted with an utterance generated by a speaker (e.g., right of o_2), and asked to click on an object (shown by the red arrow).

4.6.1 Setup

We collected 43 scenes (rooms) from the Google Sketchup 3D Warehouse, each containing an average of 22 objects (household items and pieces of furniture arranged in a natural configuration). For each object o in a scene, we create a *scenario*, which represents an instance of the communication game with o as the target object. There are a total of 2,860 scenarios, which we split evenly into a training set (denoted TR) and a test set (denoted TS). We train our learned models using the setup presented in Chapter 2, we evaluate our speaker models by creating a separate Amazon Mechanical Turk listener task, shown in Figure 4.7.

Listener Task In this task, human annotators play the role of listeners. Given an utterance generated by a speaker (human or not), the human listener must guess the target object that the speaker saw by clicking on an object. The purpose of the listener task is to evaluate speakers, as described in the next section.

4.6.2 Evaluation

Utility (Communicative Success) We primarily evaluate a speaker by its ability to communicate successfully with a human listener. For each test scenario, we asked three listeners to guess the object. We use $p_{\text{L:HUMAN}}(g \mid w)$ to denote the distribution over guessed objects g given prompt w. For example, if two of the three listeners guessed $\texttt{O}_1$, then $p_{\text{L:HUMAN}}(\texttt{O}_1 \mid w) = \frac{2}{3}$. The expected utility (4.2) is then computed by averaging the utility (communicative success) over the test scenarios TS:

$$\begin{aligned} \textsc{Success}(\textsc{s}) &= \text{EU}(\textsc{s}, \textsc{l:Human}) \\ &= \frac{1}{|\text{TS}|} \sum_{o \in \text{TS}} \sum_{w} p_{\textsc{s}}(w|o) p_{\text{L:HUMAN}}(o|w). \end{aligned} \tag{4.8}$$

Exact Match As a secondary evaluation metric, we also measure the ability of our speaker to exactly match an utterance produced by a human speaker. Note that since there are many ways of describing an object, exact match is neither necessary nor sufficient for successful communication.

We asked three human speakers to each produce an utterance w given a target o. We use $p_{\text{S:HUMAN}}(w \mid o)$ to denote this distribution; for example, $p_{\text{S:HUMAN}}(\textit{right of } \texttt{O}_2 \mid o) = \frac{1}{3}$ if exactly one of the three speakers uttered *right of* $\texttt{O}_2$. We then define the *exact match* of a speaker S as follows:

$$\textsc{Match}(\textsc{s}) = \frac{1}{|\text{TS}|} \sum_{o \in \text{TS}} \sum_{w} p_{\text{S:HUMAN}}(w \mid o) p_{\textsc{s}}(w \mid o). \tag{4.9}$$

4.6.3 Reflex versus Rational Speakers

We first evaluate speakers in the setting where only utterances of complexity 1 are allowed. Table 4.1 shows the results on both success and exact match. First, our main result is that the two rational speakers S(L:LITERAL) and S(L:LEARNED), which each model a listener explicitly, perform significantly better than the corresponding reflex speakers, both in terms of success and exact match.

Second, it is natural that the speakers that involve learning (S:LITERAL and S(L:LITERAL)) outperform the speakers that only consider the literal meaning of utterances (S:LEARNED and S(L:LEARNED)), as the former models capture subtler preferences using features.

Finally, we see that in terms of exact match, the human speaker S:HUMAN performs the best (this is not surprising because human exact match is essentially the inter-annotator agreement), but in terms of communicative success, S(L:LEARNED) achieves a higher success rate than S:HUMAN, suggesting that the game-theoretic modeling undertaken by the rational speakers is effective for communication, which is ultimate goal of language.

Speaker	Success	Exact Match
S:LITERAL [reflex]	4.62%	1.11%
S(L:LITERAL) [rational]	33.65%	2.91%
S:LEARNED [reflex]	38.36%	5.44%
S(L:LEARNED) [rational]	**52.63%**	14.03%
S:HUMAN	41.41%	**19.95%**

Table 4.1: Comparison of various speakers on communicative success and exact match, where only utterances of complexity 1 are allowed. The rational speakers (with respect to both the literal listener L:LITERAL *and the learned listener* L:LEARNED*) perform better than their reflex counterparts. While the human speaker (composed of three people) has higher exact match (it is better at mimicking itself), the rational speaker* S(L:LEARNED) *actually achieves higher communicative success than the human listener.*

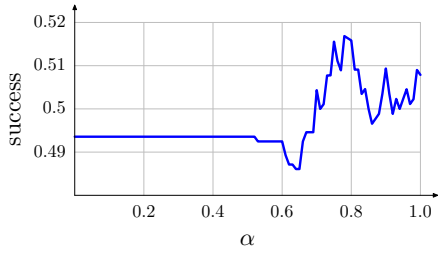


Figure 4.8: Communicative success as a function of focus parameter α without tabooing on TSDEV*. The optimal value of α is obtained at 0.79.*

Note that exact match is low even for the "human speaker", since there are often many equally good ways to evoke an object. At the same time, the success rates for all speakers are rather low, reflecting the fundamental difficulty of the setting: sometimes it is impossible to unambiguously evoke the target object via short utterances. In the next section, we show that we can improve the success rate by allowing the speakers to generate more complex utterances.

4.6.4 Generating More Complex Utterances

We now evaluate the rational speaker S(L:LEARNED) when it is allowed to generate utterances of complexity 1 or 2. Recall from Section 4.5.3 that the speaker depends on a focus parameter α, which governs the embedded listener's ability to interpret the utterance.

Taboo Amount	Success ($\alpha \to 0$)	Success ($\alpha = 1$)	Success ($\alpha = \alpha^*$)	α^*
0%	51.78%	50.99%	54.53%	0.79
5%	38.75%	40.83%	43.12%	0.89
10%	29.57%	29.69%	30.30%	0.80
30%	12.40%	13.04%	12.98%	0.81

Table 4.2: Communicative success (on TsFinal*) of the rational speaker* s(l:Learned) *for various values of α across different taboo amounts. When the taboo amount is small, small values of α lead to higher success rates. As the taboo amount increases, larger values of α (resulting in more complex utterances) are better.*

We divided the test set (Ts) in two halves: TsDev, which we used to tune the value of α and TsFinal, which we used to evaluate success rates.

Figure 4.8 shows the communicative success as a function of α on TsDev. When α is small, the embedded listener is confused more easily by more complex utterances; therefore the speaker tends to choose mostly utterances of complexity 1. As α increases, the utterances increase in complexity, as does the success rate. However, when α approaches 1, the utterances are too complex and the success rate decreases. The dependence between α and average utterance complexity is shown in Figure 4.9.

Table 4.2 shows the success rates on TsFinal for $\alpha \to 0$ (all utterances have complexity 1), $\alpha = 1$ (all utterances have complexity 2), and α tuned to maximize the success rate based on TsDev. Setting α in this manner allows us to effectively balance complexity and ambiguity, resulting in an improvement in the success rate.

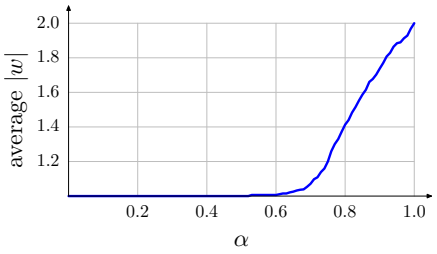


Figure 4.9: Average utterance complexity as a function of the focus parameter α on TsDev*. Higher values of α yield more complex utterances.*

4.7 Conclusion

Starting with the view that the purpose of language is successful communication, we developed a game-theoretic model in which a *rational* speaker generates utterances by explicitly taking the listener into account. On the task of generating spatial descriptions, we showed the rational speaker substantially outperforms a baseline reflex speaker that does not have an embedded model even though both models had access to the same sets of features. Our results therefore suggest that a model of the pragmatics of communication is an important factor to consider for generation.

Models which directly predict utterances, like our S:LEARNED models, conflate the contributions of semantic effects (which things are true) and pragmatic effects (which true things are best to say). Our results show that an explicit separation of a game-theoretic pragmatics, as in our S(L:LEARNED) model, can give rise to substantial gains in communicative success. In particular, this separation allows the predicted consequences of utterances, rather than just their form, to determine their utility. Pragmatic effects are especially difficult to capture in the compositional case; there, a system must manage both the structure of the nested game as well as the structure of the utterances themselves.

Our results match the intuitive sense of how successful utterances are made; however, there is still a lot of room to expand the model presented in this chapter. Specifically, we did not explore how this approach generalizes to generating sentences with rich structure. We limited the lengths of the sentences to only contain at most two spatial relations, and we restricted the system to use object ids to refer to objects, rather than using the more natural nouns, nominal descriptions, and pronouns.

A couple of issues arise when allowing for more complex descriptions that can contain pronouns. First of all, the search space over possible utterances grows exponentially in the length of the sentence. The brute force algorithm presented here to search the space of utterances does not scale. Secondly, we must address the question of why people prefer to use pronouns to refer rather than using names or definite descriptions. Pronouns introduce an element of ambiguity about which antecedent they denote, which can result in a higher chance of interpretation failure. Intuitively, people prefer to use pronouns because they are less complex. To accurately model the trade off between complexity and communication success, one needs a more nuanced measure of utterance complexity than presented here.

Chapter 5

Grounding Spatial Relations for Human-Robot Interaction

5.1 Introduction

In this chapter, we present a natural language interface for interacting with a robot that allows users to issue commands and ask queries about the spatial configuration of objects in a shared environment. To accomplish this goal, the robot must interpret the natural language sentence by grounding it in the data streaming from its sensors. Upon understanding the sentence, the robot then must produce an appropriate response via action in the case of a command, or via natural language in the case of a query.

For example, to correctly interpret and execute the command "Pick up the cup that is close to the robot" (see Fig. 5.1) the system must carry out the following steps: (i) ground the nouns (e.g. "cup") in the sentence to objects in the environment via percepts generated by the robot's sensors; (ii) ground the prepositions (e.g. "close to") in the sentence to relations between objects in the robot's environment; (iii) combine the meanings of the nouns and prepositions to determine the meaning of the command as a whole; and (iv) robustly execute a set of movements (e.g. PickUp) to accomplish the given task.

In order for a robot to effectively interact with a human in a shared environment, the robot must be able to recognize the objects in the environment as well as be able to understand the spatial relations that hold between these objects. The importance of interpreting spatial relations is evidenced by the long history of research in this area (Burgard et al., 1999; Skubic et al., 2004; Moratz and Tenbrink, 2006; Kelleher et al., 2006; Zender et al., 2009). However, most of the previous work builds models of spatial relations by hand-coding the meanings of the spatial relations rather than learning these meanings from data. One

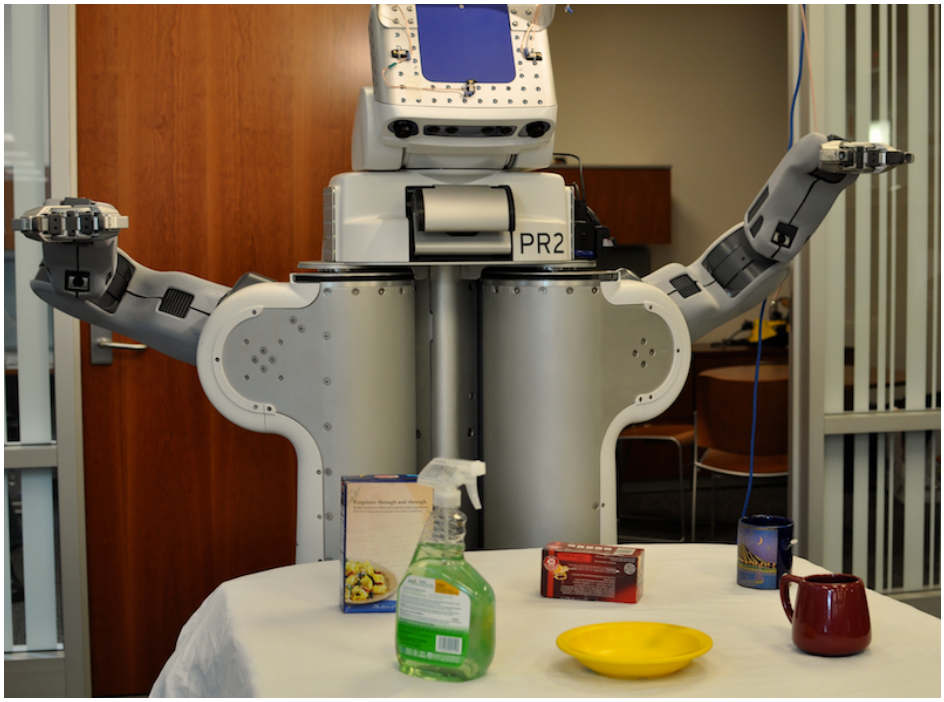


Figure 5.1: An example of the visual setting in which the PR2 robot is issued commands and asked queries.

of the conclusions presented in Golland et al. (2010) is that a learned model of prepositions can outperform one that is hand-coded. In the present work, we extend the learned spatial relations models presented in Golland et al. (2010) to handle a broader range of natural language (see Table 5.1) and to run on a PR2 robot in a real environment such as the one in Fig. 5.1.

The spatial relations model presented in Golland et al. (2010) had several limitations that prevented it from being deployed on an actual robot. First, the model assumed perfect visual information consisting of a virtual $3D$ environment with perfect object segmentation. Second, the model only allowed reference to objects via object ID (e.g. $\mathtt{O}_3$) as opposed to the more natural noun reference ("the cup"). Lastly, the grammar was small and brittle, which caused the system to fail to parse on all but a few carefully constructed expressions. In this work, we extend the model in Golland et al. (2010) to address these limitations by building a system that runs on a PR2 robot and interacts with physical objects in the real-world. In order to interpret the sentences in Table 5.1, we have built the following modules:

- A vision module that provides grounding between visual percepts and nouns (section 5.3.2)
- A spatial prepositions module capable of understanding complex $3D$ spatial relationships between objects (section 5.3.3)
- A set of actions implemented on a PR2 robot to carry out commands issued in natural language (section 5.3.4)

We have created an integrated architecture (see Fig. 5.2), that combines and handles the flow of information of the separate modules. The system is managed by an interface where

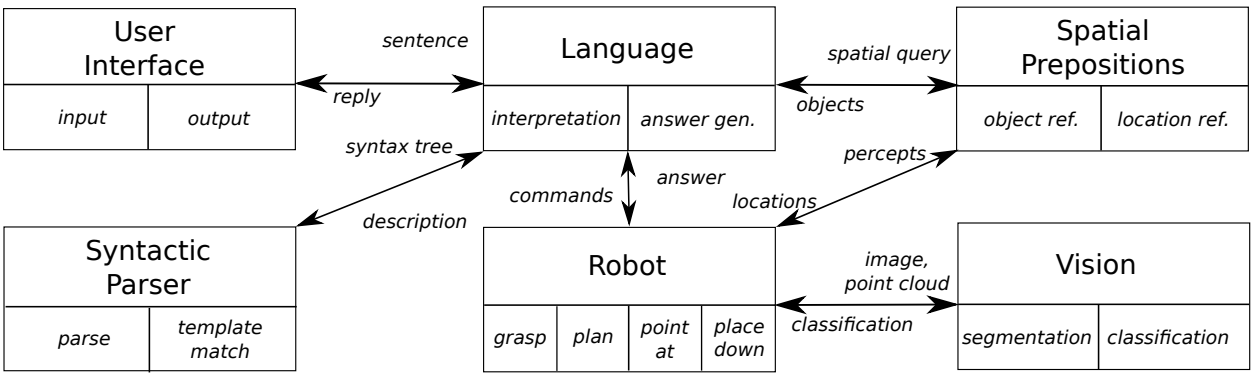


Figure 5.2: The architecture of our system showing the interactions between the modules.

a user types sentences, and the robot replies either by answering questions or executing commands (see Table 5.1). Every sentence is semantically analyzed to determine both the type of query or command as well as the identity of all objects or locations referenced by the sentence. The semantic interpretation depends on the vision module to interpret the nouns and on the prepositions module to interpret the spatial relations present in the sentence. If the sentence issued by the user is interpreted as a command, then the appropriate action and parameters are sent to the robot module. The results from the queries and feedback from the action's execution are finally displayed on the user interface.

5.2 Related Work

Natural language understanding and grounding has been studied since the beginning of artificial intelligence research, and there is a rich literature of related work. Recently, the availability of robotic agents has opened new perspectives in language acquisition and grounding. In a seminal work, Steels and Vogt (1997) studied the emergence of language among robots through games. While we retain some of the ideas and concepts, the main difference between our approach and Steels' is that we provide the robot with the vocabulary, whereas in Steels and Vogt (1997) the perceptual categories arise from the agent out of the game strategy. In a similar fashion Roy (2002) developed a model that could learn a basic syntax and ground symbols to the sensory data.

Kuipers (2000) introduced the idea of Spatial Semantic Hierarchy (SSH), where the environment surrounding the robot is represented at different levels, from geometric to topological. An extension of this work is in MacMahon et al. (2006), where the authors develop a system that follows route instructions. The main contribution is in the automatic synthesis of implicit commands which significantly improves the robot's performance. However the proposed approach uses fixed rules rather than learning the spatial relationships from data. In our work we claim that learning models from data is a key component towards natural communication with a robot.

A different approach is to teach language to robots as they perceive their environment. For example, Nakano et al. (2010) present an approach where robots ground lexical knowledge through human-robot dialogues where a robot can ask questions to reduce ambiguity. A more

natural approach was presented in Iwahashi et al. (2010), where the robot learns words for colors and object instances through physical interaction with its environment. Whereas the language used in Iwahashi et al. (2010) is very simple and static, our approach uses complex language that supports spatial reference between objects.

Given the relevance of spatial relations to human-robotic interaction, various models of spatial semantics have been proposed. However, many of these models were either hand-coded (Burgard et al., 1999; Moratz and Tenbrink, 2006) or in the case of Skubic et al. (2004) use a histogram of forces (Matsakis and Wendling, 1999) for $2D$ spatial relations. In contrast, we build models of $3D$ spatial relations learned from crowd-sourced data by extending previous work (Golland et al., 2010).

Some studies consider dynamic spatial relations. In Matuszek et al. (2010), a robot must navigate through an office building, thereby parsing sentences and labeling a map using a probabilistic framework. In Chen and Mooney (2011), a simulated robot must interpret a set of commands to navigate throughout a maze. Our current work focuses mainly on understanding complex spatial relationship between static objects.

Tellex et al. (2011a) investigates learning a language grounding, where a robotic forklift

Input	**Action**
"**What is** the object in front of PR2?"	REPLY("A tea box")
"**Which object is** the cup?"	REPLY("It is $\texttt{O}_3$")
"**Which object is** behind the item that is to the right of the cup?"	REPLY("It is $\texttt{O}_7$")
"**Which object is** close to the item that is to the left of the green_works?"	REPLY("It is $\texttt{O}_6$")
"**Point at the area** on the plate."	POINTAT([XYZ])
"**Point to** the object to the left of the tea_box."	POINTTO($\texttt{O}_3$)
"**Place** the cup **in the area** behind the plate."	PLACEAT($\texttt{O}_3$, [XYZ])
"**Place** the pasta_box **in the area** on the plate."	PLACEAT($\texttt{O}_4$, [XYZ])
"**Pick up** the cup that is far from the robot."	PICKUP($\texttt{O}_6$)
"**Put down** the cup **in the area** inside the bowl."	PLACEAT($\texttt{O}_6$, [XYZ])
"**Pickup** the tea_box in front of the plate."	PICKUP($\texttt{O}_2$)
"**Put down** the object **in the area** near to the green_works and far from you."	PLACEAT($\texttt{O}_2$, [XYZ])
"**Move** the object that is near to the robot **to the area** far from the robot."	MOVETO($\texttt{O}_2$, [XYZ])
"**Move** the cup close to the robot **to the area** in front of the plate and behind the tea_box."	MOVETO($\texttt{O}_3$, [XYZ])

Table 5.1: Examples of sentences handled by our system and the corresponding interpretation.

receives commands via natural language. The grounding is obtained by dynamically instantiating a probabilistic graphical model that links parts of a sentence to percepts in the real world. In contrast, our work assumes that this grounding has already been learnt (via offline training). However, our language module is capable of handling more complex sentences than Tellex et al. (2011a), such as those in Table 5.1. We are currently investigating methods of learning the grounding via generative models without losing the ability to handle arbitrarily complex spatial descriptions.

In this chapter we focus mainly on grounding nouns and spatial prepositions while keeping a predefined set of actions the robot can perform. We are working towards more flexible commands to allow the robot to execute high-level actions as in Branavan et al. (2010) without requiring the user to specify all the intermediate steps required to accomplish the goal.

5.3 System Description

We describe how the language, vision, spatial prepositions, and robotics modules operate to enable the robot system to respond to natural language commands in a visual environment.

5.3.1 Language Module

The language module takes as input a textual, natural language utterance $\mathcal{U}$, which can contain instructions, references to objects either by name or description (e.g. "plate" or "the cup close to the robot"), and descriptions of spatial locations in relation to other objects (e.g. "area behind the plate"). The output of the language module is a command $\mathcal{C}$ to the robot containing the interpretation of the utterance (e.g. PICKUP($\mathtt{O}_4$)). Interpreting $\mathcal{U}$ into $\mathcal{C}$ happens in three steps: template matching, which decides the coarse form of the sentence; broad syntactic parsing, which analyzes the structure of the sentence; and deep semantic analysis which interprets the linguistic sentence in terms of concepts in the visual setting.

Template Matching First, the utterance $\mathcal{U}$ is matched against a list of manually constructed templates. Each template specifies a set of keywords that must match in $\mathcal{U}$, as well as gaps which capture arbitrary text spans to be analyzed in later steps (a subset are shown in Table 5.1 with keywords shown in bold).[1] Each template specifies the query or command as well as which spans of $\mathcal{U}$ correspond to the object descriptions referenced in that command. For example, in the utterance "pick up the cup that is close to the robot", the template would match the keywords "pick up" and triggers a PICKUP command to send to the robot. The text spans that must be interpreted as object ids or locations in the

[1]These templates were constructed based only on the development data.

environment (such as "the cup that is close to the robot" in our example) are passed to the second step for deeper interpretation.

Although theoretically this template approach structurally limits the supported commands and queries, the approach still covers many of the phenomena present in our data. During evaluation, the templates covered 98% of the tested sentences (see Table 5.3), despite the fact that the humans who generated these sentences were not aware of the exact form of the templates and only knew the general set of actions supported by the robot. We employ the template approach because it closely matches the pattern of language that naturally arises when issuing commands to a robot with a restricted scope of supported actions. Rather than focusing on a broad range of linguistic coverage that extends beyond the capabilities of the robot actions, we focus on deep analysis. In the second and third steps of linguistic interpretation (described below) our system does model recursive descriptions (e.g. "the book on the left of the table on the right of the box"), which are the main linguistic complexity of interest.

Broad Syntactic Parsing In order to robustly support arbitrary references to objects and locations, we parse these descriptions $\mathcal{R}$ with a broad-coverage syntactic parser (Petrov and Klein, 2007) and then use tree rewrite rules to project the output syntactic parse onto our semantic grammar $\mathcal{G}$:

[noun]	N	$\rightarrow$	`plate` | `cup` | $\cdots$
[preposition]	R	$\rightarrow$	`close_to` | `on` | $\cdots$
[conjunction]	NP	$\rightarrow$	N RP*
[relativization]	RP	$\rightarrow$	R NP

We apply the following tree rewrite rules to normalize the resulting tree into $\mathcal{G}$:[2]

- rename preposition-related POS tags (IN, TO, RB) to P
- crop all subtrees that fall outside $\mathcal{G}$
- merge subtrees from multi-word prepositions into a single node (e.g. "to the left of" into "left")
- to handle typos in the input, we replace unknown prepositions and nouns with those from the lexicons contained in the preposition and vision modules that are closest in edit-distance, provided the distance does not exceed 2

Deep Semantic Analysis The last step of interpretation takes as input a tree $\mathcal{T}$ from our semantic grammar that either refers to a specific object in the robot's environment or a specific $3D$ location. The deep semantic analysis returns the corresponding object id or a list of $3D$ points. For example, in the case of object reference, this step would take the description "the cup that is close to the robot" and return object id $\texttt{O}_4$ (see Fig. 5.7). We follow the method of probabilistic compositional semantics introduced in Golland et al.

[2]These rules were manually generated by analyzing the development data.

(2010) to compute a distribution over objects $p(o|\mathcal{R})$ and return the object id that maximizes $\arg\max_o p(o|\mathcal{R})$. Concretely, $\mathcal{T}$ is recursively interpreted to construct a probability distribution over objects. We follow the semantic composition rules presented in Golland et al. (2010) at all subtrees except those rooted at N. If the subtree is rooted at N with noun child w, we attain a distribution over objects by leveraging object recognition model (section 5.3.2). We use Bayesian inversion with the uniform prior to transform the object recognition distribution $p(w|o)$ into a distribution over objects given the noun: $p(o|w)$. If the subtree is rooted at RP with children R and NP, the interpretation calls out to the prepositions module (section 5.3.3) to attain the distribution over objects (or $3D$ points, in the case of a location reference) that are in relation R to each of the objects in the recursively computed distribution NP.

5.3.2 Vision Module

The role of the vision module is twofold: (i) segment the visual input captured by a $3D$ Asus Xtion RGB image and point cloud and (ii) assign a classification score between a noun N and an object id that corresponds to how well the noun describes the object.

Training

We trained our object classifier with 50 objects, mainly kitchen and office objects. To obtain training images, we placed the object on a turning table and collected images at a frequency of about 10° per image, collecting around 80 images per object class. Following the idea of Winder and Brown (2007), we introduced jittering effects to the objects to make the classifier robust against view and perspective changes. Specifically, after we cropped the object inside the bounding box, we randomly transposed, rotated, and scaled the bounding boxes.

Segmentation

The $3D$ point cloud captured by the camera is voxelized at a resolution of $1mm$ to reduce the number of points. The points generated from voxelization are transformed from the camera into the robot frame of reference, using the kinematic chain data from the PR2 robot. We fit the plane of the tabletop by applying RANSAC. We constrained the RANSAC by assuming that the table is almost parallel to the ground. All the points that do not belong to the table are clustered to segment out tabletop objects. Noise is reduced by assuming that each object must have a minimum size of $3cm$. The point cloud clusters are subsequently projected into the image to identify image regions to send to the classification module. Fig. 5.3 shows a segmentation example as described above.

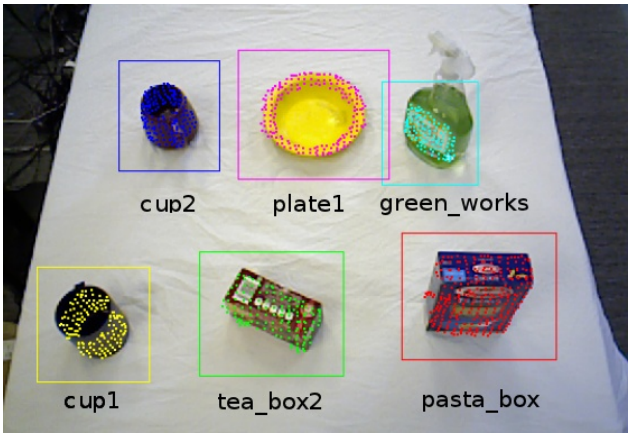


Figure 5.3: View of scene in Fig. 5.1 from the camera perspective. Segmented objects are enframed, corresponding point cloud points are depicted, and object labels are shown.

Classification

Often, the segmentation component produces well-centered object bounding boxes, allowing us to directly perform object classification on bounding boxes instead of performing object detection, e.g., with a slower sliding window based approach. We apply the standard, state-of-the-art image classification algorithms using features extracted by a two-level pipeline, (i) the coding level densely extracts local image descriptors, and encodes them into a sparse high-dimensional representation and (ii) the pooling level aggregates statistics in specific regular grids to provide invariance to small displacement and distortions. We use a linear SVM to learn the parameters and perform the final classification.

Specifically, we perform feature extraction using the pipeline proposed in Coates et al. (2010). This method has been shown to perform well with small to medium image resolutions,[3] and it is able to use color information (which empirically serves as an important clue in instance recognition). Additionally, the feature extraction pipeline runs at high speed because most of its operations only involve feed-forward, convolution-type operations. To compute features, we resized each bounding box to 32×32 pixels, and densely extracted 6×6 local color patches. We encoded these patches with ZCA whitening followed by a threshold encoding $\alpha = 0.25$ and a codebook of size 200 learned with Orthogonal Matching Pursuit (OMP). The encoded features are max pooled over a 4×4 regular grid, and then fed to a linear SVM to predict the final label of the object. Feature extraction has been carried out in an unsupervised fashion, allowing us to perform easy retraining, should new objects need to be recognized. Fig. 5.4 illustrates the key components of our pipeline, and we defer to Coates et al. (2010) for a detailed description.

[3] Our RGB+depth images have resolution 640×480.

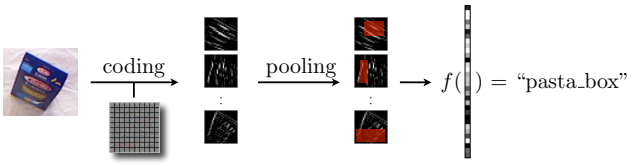


Figure 5.4: The classification pipeline adopted to train object classifiers.

5.3.3 Spatial Prepositions Module

Given a preposition and landmark object, the prepositions module outputs a distribution over the target objects and $3D$ points that are located in the given preposition in relation to the given landmark object from the robot's point of view.[4] For these experiments, we use the Hybrid Model presented in Section 2.4.

Interpreting Object References

The relativization rule in the grammar $\mathcal{G}$ relies on the Hybrid Model of spatial prepositions in order to refer to objects by their physical locations. For example, to interpret the sentence "Pick up the cup that is close to the robot" the language module prompts the prepositions module for a distribution over objects $p(g|\texttt{close\_to}, \texttt{O}_{robot})$, where $\texttt{O}_{robot}$ corresponds to the object id assigned to the robot ($\texttt{O}_1$ in Fig. 5.7).

Interpreting Location References

To interpret references to locations like "the area on plate" or "the area in front of the plate and behind the tea_box" the system returns a distribution over $3D$ points that fall in the described areas.

We simulate placing the target object BB in 1,000 random positions within the boundaries of the table and compute the likelihood of each position given the set of spatial relations expressed in the location reference. We return the best 50 locations, to be filtered by the robot planner to avoid collisions.

For example, "on the plate" refers to the area on $\texttt{O}_7$ (see Fig. 5.5) and "behind the plate" refers to the area behind $\texttt{O}_7$ (see Fig. 5.6). While the description "in front of the plate and behind the tea_box" refers to the intersection of the area in front of $\texttt{O}_7$ and behind $\texttt{O}_2$ (see Fig. 5.7).

[4]We only consider the robot's point of view for all spatial references.

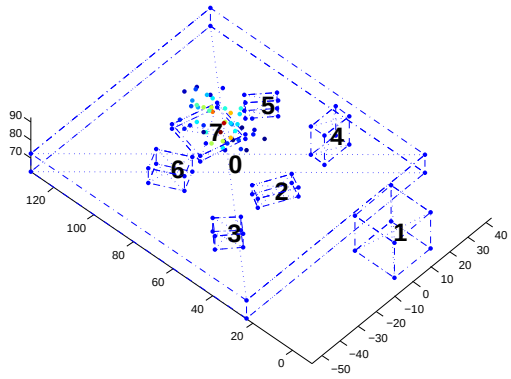


Figure 5.5: "on the plate"

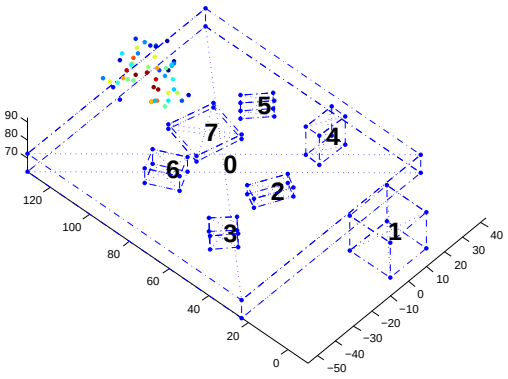


Figure 5.6: "behind the plate"

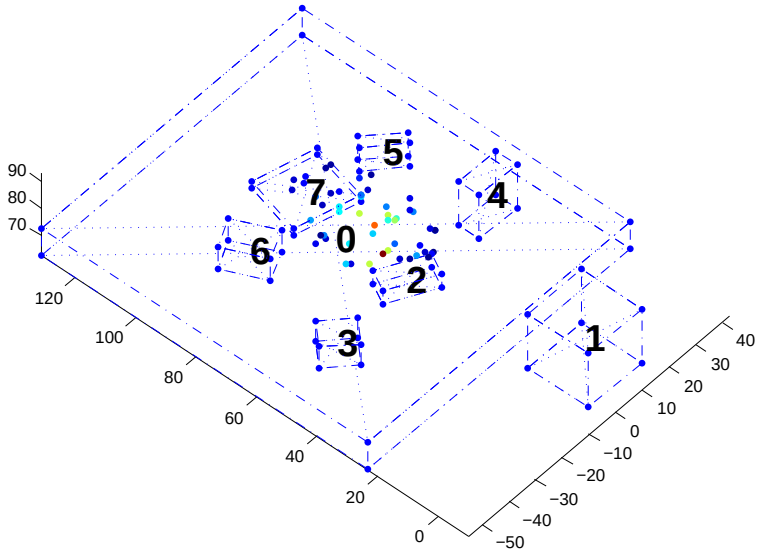


Figure 5.7: "in front of the plate and behind the tea_box"

$3D$ points satisfying various spatial relations with regards to the `tea_box` (O_2), and `plate` (O_7) in the scene depicted in Fig. 5.1.

5.3.4 Robotic Module

Our robotic platform is a mobile manipulator PR2 robot manufactured by Willow Garage.[5] It is two-armed with an omni-directional driving system. Each arm has 7 degrees of freedom. The torso has an additional degree of freedom as it can move vertically. The PR2 has a variety of sensors, among them a tilting laser mounted in the upper body and a $3D$ Asus Xtion camera over a pan-tilt head. During our experiments we used the tilting laser to create a static $3D$ map of the robot's surroundings, and we used the Asus camera to segment and recognize the objects. For planing and executing a collision-free trajectories with the $7DOF$ arms (Şucan et al., 2012), we used a compact $3D$ map (Wurm et al., 2010).

[5] `http://www.willowgarage.com`

The supported robot actions include:

- PickUp(O): Pick up object O using the algorithm presented in Hsiao et al. (2010).
- PointTo(O): Point to object O.
- PointAt([XYZ]): Point at location [XYZ].
- PlaceAt([XYZ]): Place down a grasped object in location [XYZ].
- MoveTo(O,[XYZ]): Move object O into location [XYZ].

PickUp was implemented by using the object's $3D$ point cloud to compute a good grasping position and by planning a collision-free trajectory to position the gripper for grasping.

PointTo was implemented by moving the robot's gripper so that its tool frame points at the centroid of the object's point cloud. Several candidate gripper positions are uniformly sampled from spheres of various radii around the object. The gripper orientation is chosen to be an orthogonal basis of the pointing vector. The first candidate gripper pose that has a collision-free trajectory is selected for execution.

The PlaceAt action takes as input a candidate list of scored $3D$ points generated by the spatial prepositions module 5.3.3, and places an object held in the gripper at one of these $3D$ points. The robot executes the PlaceAt action using the highest scoring candidate PlaceAt point that yields a collision-free trajectory for the gripper.[6] The exact location for placing takes into account the gripper shape and the object height.

The MoveTo action is implemented by combining PickUp and PlaceAt.

5.4 Experimental Results

We describe the results of the independent modules, as well the end-to-end accuracy of the entire system.

5.4.1 Experimental Scenario

During our experiments we have collected 290 utterances during development and 210 utterances during the online test. Each utterance is either a command or query issued to the robot by the user in a shared visual setting. As mentioned in section 5.3, we use the development set to design the templates in the language module and perform model selection in the spatial prepositions module. We additionally have collected separate offline datasets in order to train our object classification and preposition models. To train the preposition model, we use the $3D$ virtual environment dataset collected in Golland et al. (2010) composed

[6] To support stacking objects or placing one inside another, we allowed collisions between the gripper-held object and the environment.

by 43 rooms and consisting of 2,860 tuples of the form (virtual environment, target object, preposition, reference object). To train the object classification model, we collect a dataset of 80 pairs of (image, instance label) for each of the 50 possible objects, totalling 4,000 labelled images, that appear in our visual settings. Parameter selection for the remaining components was done using the development set of command/query and visual setting. The result of the robot's execution in response to a command, or linguistic answer in response to a query are evaluated to be either correct or incorrect. We report the performance of each independently trained module (either coverage or accuracy) as well as the accuracy of the overall system on the online test set.

A typical testing scenario is shown in Fig. 5.1. The dominant feature of the robot's environment is a flat tabletop covered with a set of objects with which the robot will interact. Although we used a white sheet to cover the table, none of our modules depend on a specific background color. Since the objects are segmented using $3D$ point clouds, we assume they are placed at least $2cm$ apart. We further assume all objects are visible by the robot without changing its pan-tilt head configuration, and are reachable by at least one arm without having to move the holonomic base.

5.4.2 Vision Results

The object classifier is able to recognize the objects at a high accuracy. In our offline testing, the classifier achieves a 99.8% one-vs-all 10-fold cross validation accuracy over the training set.

In the online testing experiments, the robot was using real data, looking from a 70° angle at the objects on the table. We measured two different accuracies, first the accuracy of the object segmentation, which has to find the patches in the images that contain the objects, and second the classification of the segmented image patch. The object segmentation achieves an accuracy of 94% when we evaluate the segmentation over the objects contained in the online testing. Wrong object segmentations were usually caused by reflective object surfaces, e.g., reflective pans. Here, the Asus camera did not perceive $3D$-points in larger areas of the objects. Typically, in these cases the object segmentation identified more than one object.

The object classification achieved an accuracy of 91.7% during online testing. Only correctly segmented object areas were considered for this classification experiment. However, in this work we are more interested in the selection task than classification. In the selection task, the goal is to select the object o_w with the highest likelihood for given a class w: $o_w = \text{argmax}_o(p(o|w))$, while in the classification task the goal is to select the object class w_o with the highest likelihood given the object o: $w_o = \text{argmax}_w(p(w|o))$. When the task is to select one object among the ones on the table, the selection accuracy is 97%. The selection accuracy is higher that the classification accuracy mainly because the model has to choose between 5-6 objects, while for the classification the model has to choose between 50 labels.

Segmentation	Classification	Selection
94%	91.7%	97%

Table 5.2: Vision Results in the Online Test

Task	Accuracy
Template Matching	98%
Grammar Coverage	94%
Noun Interpretation	97%
Preposition Interpretation	95%
Sentence Interpretation	91%
Valid Execution	89%

Table 5.3: Overall Results in the Online Test

5.4.3 Overall Results and Error Analysis

To evaluate the performance of the whole system we measured the accuracy of each of the needed steps to correctly interpret and answer/execute a question/command given in natural language by the user. The results for each of these steps on the online test set (210 sentences) are presented in Table 5.3:[7]

- Template Matching: Percentage of sentences that match one of the predefined templates. In this case the main source of errors is the user misspelling of words (a).
- Grammar Coverage: Percentage of sentences that the Language Module can parse after the template matching succeed. In this case the main source of errors are failures in the tree normalization process (b,c).
- Noun Interpretation: Percentage of nouns that the language module can generate a valid answer using the classification results from the vision module. In this case the main source of errors is the wrong segmentation (d,e).
- Preposition Interpretation: Percentage of spatial prepositions that the language module can generate a valid answer using the predictions results from the spatial preposition module. In this case the main source of error is the ambiguity of the target referred by the spatial prepositions model (f,g).
- Sentence Interpretation: Percentage of parsed sentences that the system can correctly interpret by combining the results from the noun and preposition interpretations according to the syntactically normalized tree. In this case the main source of error is the inability of the system to choose the best answer within the valid set of answers (e,g,h).
- Execution: Percentage of interpreted sentences that the robot can execute correctly (when the input is a command). The main sources of errors are non-reachable poses in the robot's configuration space and collisions during placing (i,j).

[7] In parentheses we have included references to examples of errors from Table 5.4.

(**a**) *Poit* the left of the bowl (Template)

(**b**) **Which object is** behind the item which is to the left of the cup? (Grammar)

(**c**) **Pickup** the cup near to PR2 (Grammar)

(**d**) **What is** to the left of the pan (Nouns)

(**e**) **Place** the tea_box **in the area** near to the coffe_mate (Nouns)

(**f**) **Point at** the object on the left of the green_works (Prepositions)

(**g**) **Point to** the object to the left of the tea_box (Prepositions)

(**h**) **Which object is** to the left of the mug and to the right of the cup? (Sentence)

(**i**) **Pick up** the pan (Execution)

(**j**) **Move** the cup in front of the pan **into the area** on the clock (Execution)

Table 5.4: Examples of Failed Sentences

5.5 Discussion and Conclusions

The contribution of this chapter is an extension of Golland et al. (2010) to a real robotic system with sensor-driven perception for grounding nouns and spatial relations. It is noteworthy that, while the data used for training the spatial prepositions module has been acquired via a virtual world, the model has proven general enough to yield acceptable performance in a real robotics scenario.

Our results in Table 5.3 show that the overall system is capable of executing complex commands issued in natural language that grounds into robotic percepts. Although the modules in our system are trained in isolation, a correct interpretation requires that they all work together.

Our results suggest that stronger integration between modules is a fruitful avenue for reducing interpretation errors. For instance, the combination of the noun interpretation with preposition interpretation helps to reduce ambiguity in the descriptions. For example, in Fig. 5.3 there are two cups and three objects close to the robot, and therefore the commands "pick up the cup" and "pick up the object close to the robot" are ambiguous. However, the command "pick up the cup close to the robot" helps determine the relevant cup. This capability could be used to enable multiple grounding sources for the objects. For example,

assertions like "the object in front of the plate is a tea_box" or "the green_works is the object behind the pasta_box" can be used to teach the system new labels and spatial relations via linguistic input.

Stronger integration between components within a single module can also help reduce errors. Currently, the language interpretation module works in a feed-forward, pipelined approach: first templates are used for coarse language matching, next text spans are parsed and projected into a small semantic grammar, and then the semantic trees are interpreted. A failure in one layer will propagate to subsequent layers. In future work, we plan to refactor our model to remove this limitation by performing joint inference so that decisions are made using information from all steps of the process. We eventually plan to extend the joint model to incorporate the computer vision and spatial prepositions module, so that all components share information more directly in order to help each other make decisions.

Our system correctly interprets many of the input sentences; therefore, in addition to reducing the errors, we are interested in extending the system to handle increased complexity. We are working towards enabling the robot to understand and execute more complex sentences, including actions that will require a degree of planning or actions that unfold over long periods of time. Moreover, we are working to grow the lexicon beyond our initial set of nouns and prepositions. We are additionally working on enabling the robot to operate in more generic, non-tabletop scenarios.

Chapter 6

Conclusion

6.1 Challenges and Findings

The question of how language conveys meaning has been studied as far back as Frege (1879). Frege introduced the compositional approach to modeling meaning: first determine the meaning of the words and phrases, and then combine them using a set of rules to determine the meaning of the sentence as a whole. The elegant simplicity in his approach is the uniformity of the composition rules: the meaning of nearly every sentence is can be determined using the same set of rules. However, the hidden complexity in Frege's approach is the tremendous variety in the meanings of the elemental words and phrases. In order to apply Frege's compositional approach to ground the meaning of linguistic expressions in the real world, one must make explicit these elemental lexical meanings.

In this thesis, we present a model of lexical semantics, restricted to spatial language. We present how we can model the meaning of various prepositions in order to determine whether the relations they denote hold true of a set of objects arranged in space. Even in the narrow domain of spatial language, we encounter challenges in building a model of lexical semantics. Spatial relations are inherently vague, their meaning depends on the context, and, even in this restricted setting, binary spatial relations are rich enough that they can be recursively combined to construct arbitrarily long descriptions. We have shown how our lexical model of binary spatial relations can be compositionally applied to automatically interpret of a wide range of utterances that vary in length and complexity.

After Frege, Austin (1962) posited that language was not simply about expressing meaning, but is instead for doing things. In this view, language is seen as a tool used by people to cooperatively accomplish goals. As opposed to Frege, viewing language as a tool is a shift in perspective from the listener — whose goal is to interpret the meaning of a given sentence — to the speaker — who is instead interested in determining which sentence to utter in the first

place. Using ideas from Grice (1975), we have shown how to formalize the view of language as a cooperative tool into a system that selects utterances to communicate successfully with a listener. Our model is able to generate a reference to an object using an expression that is both truthful and unambiguous.

6.2 Future Work

In this work, we have made several simplifying assumptions and have limited our focus to interpreting spatial descriptions that refer to objects. It would be exciting to see our model extended to interpret references containing expressions beyond spatial descriptions (e.g. adjectives describing color, shape, size, etc.) or to more complex interactions, such as extended dialogue systems.

We have restricted our study to a small, fixed set of spatial prepositions, and even in this small set, there is a lot left to explore. We explicitly chose not to model the effect of frame of reference on the meaning of projective prepositions, but there is a lot to learn about how the context determines which frame of reference is active, and how computer vision techniques can be used to classify an object's intrinsic frame of reference. There is a richer structure to the meaning of the prepositions in our list; for example, *on* can refer to a "clingy attachment" such as *stamp on envelope*, a "fixed attachment" such as *handle on door*, or an "encircle with contact" such as *ring on finger* (Feist, 2008). Moreover, the meaning of a spatial relation is not fixed, as is evidenced by the multitude of modifier phrases that change the meaning of a spatial relation (e.g. *10 meters* or *just slightly*). There are spatial relations outside the set we considered that appear to behave differently in that they refer to the arrangement (*surrounded by*) or density (*interspersed among*) of plural arguments. Still others rely on regions of space determined by the specific geometry of imagined reference objects (*in the 'V' formed by the two rivers*) occasionally by metaphor (*located on your 6 o'clock*). Overall, the range of linguistic expression and subtleties in meaning of spatial relations is enormous — we barely scratched the surface.

Another interesting area for exploration is to extend the conception of language as a cooperative-game presented in Chapter 4. Although we looked at modeling listener confusion, we did not explore how processing constraints of the speaker influence the choice of utterance. One place to start might be Wilson and Sperber (2002), who develop a computational model that balances "processing power" with the goal of "maximizing relevance." Once this issue is resolved, it would be exciting to determine other applications of the cooperative game model. For instance, one natural application would be a computational model of the trade-off between phonological contrast and articulatory effort when humans generate words (Kirchner, 2001). Yet another direction for exploration on the pragmatics front is extending the pragmatic model to capture presupposition failure or to model speech acts where there is a larger discrepancy between speaker meaning and utterance meaning than the commands and queries explored in Chapter 5.

6.3 Parting Thoughts

We hope this thesis has sparked curiosity, regardless of the reader's background. The skeptic may focus on the limitations of our approach and question how broadly our ideas generalize to new domains. The scientist, motivated by our observations, may develop questions that lead to a more conscious understanding of how language works. The engineer may consider how to apply the practical techniques we present to solving the design challenges he faces day-to-day. The philosopher may wonder about the nature of language and how it relates to mind and reality. The dreamer imagines the day when we finally give language to artifacts that will answer these questions for us.

Bibliography

Yoav Artzi and Luke Zettlemoyer. Bootstrapping semantic parsers from conversations. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2011. URL `http://www.aclweb.org/anthology/D11-1039`.

J. L. Austin. *How to do Things with Words: The William James Lectures delivered at Harvard University in 1955*. Oxford, Clarendon, UK, 1962.

David Bailey, Jerome Feldman, Srini Narayanan, and George Lakoff. Modeling embodied lexical development. In *Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society*, volume 322. Lawrence Erlbaum, 1997.

Kobus Barnard, Pinar Duygulu, David Forsyth, Nando De Freitas, David M. Blei, and Michael I. Jordan. Matching words and pictures. *The Journal of Machine Learning Research*, 3, 2003.

Anton Benz, Gerhard Jäger, and Robert Van Rooij. *Game theory and pragmatics*. Palgrave Macmillan, 2005.

Daniel Gureasko Bobrow. *Natural language input for a computer problem solving system*, volume 1. Massachusetts Institute of Technology., 1964.

Benjamin Börschinger, Bevan K. Jones, and Mark Johnson. Reducing grounded learning tasks to grammatical inference. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2011. URL `http://www.aclweb.org/anthology/D11-1131`.

S.R.K. Branavan, Harr Chen, Luke Zettlemoyer, and Regina Barzilay. Reinforcement learning for mapping instructions to actions. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. Association for Computational Linguistics, 2009. URL `http://www.aclweb.org/anthology/P/P09/P09-1010`.

S.R.K. Branavan, Luke S. Zettlemoyer, and Regina Barzilay. Reading between the lines: Learning to map high-level instructions to commands. In *In Proceedings of the Association for Computational Linguistics (ACL)*, 2010.

S.R.K. Branavan, David Silver, and Regina Barzilay. Learning to win by reading manuals in a monte-carlo framework. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 2011. URL `http://www.aclweb.org/anthology/P11-1028`.

Wolfram Burgard, Armin B. Cremers, Dieter Fox, Dirk Hähnel, Gerhard Lakemeyer, Dirk Schulz, Walter Steiner, and Sebastian Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114(1):3–55, 1999.

Laura A. Carlson-Radvansky and Gordon D. Logan. The influence of reference frame selection on spatial template construction. *Journal of memory and language*, 37(3):411–437, 1997.

David L. Chen and Raymond J. Mooney. Learning to sportscast: A test of grounded language acquisition. In *International Conference on Machine Learning (ICML)*, pages 128–135. Omnipress, 2008.

David L. Chen and Raymond J. Mooney. Learning to interpret natural language navigation instructions from observations. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI-2011)*, pages 859–865, 2011.

David L. Chen, Joohyun Kim, and Raymond J. Mooney. Training a multilingual sportscaster: Using perceptual context to learn language. *Journal of Artificial Intelligence Research*, 2010.

James Clarke, Dan Goldwasser, Ming-Wei Chang, and Dan Roth. Driving semantic parsing from the world's response. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*. Association for Computational Linguistics, 2010.

Adam Coates, Honglak Lee, and Andrew Y Ng. An analysis of single-layer networks in unsupervised feature learning. *Ann Arbor*, 1001:48109, 2010.

Fintan J. Costello and John D. Kelleher. Spatial prepositions in context: The semantics of near in the presence of distractor objects. In *Proceedings of the Third ACL-SIGSEM Workshop on Prepositions*, pages 1–8. Association for Computational Linguistics, 2006.

Kenny R. Coventry, Angelo Cangelosi, Rohanna Rajapakse, Alison Bacon, Stephen Newstead, Dan Joyce, and Lynn V Richards. Spatial prepositions and vague quantifiers: Implementing the functional geometric framework. In *Spatial Cognition IV. Reasoning, Action, Interaction*, pages 98–110. Springer, 2005.

Bob Coyne and Richard Sproat. Wordseye: an automatic text-to-scene conversion system. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 487–496. ACM, 2001.

Ioan A. Şucan, Mark Moll, and Lydia E. Kavraki. The Open Motion Planning Library. *IEEE Robotics & Automation Magazine*, 2012. URL `http://ompl.kavrakilab.org`. To appear.

Robert Dale and Nicholas Haddock. Generating referring expressions involving relations. In *Proceedings of the fifth conference on European chapter of the Association for Computational Linguistics*, pages 161–166. Association for Computational Linguistics, 1991.

Robert Dale and Ehud Reiter. Computational interpretations of the gricean maxims in the generation of referring expressions. *Cognitive Science*, 19(2):233–263, 1995.

Judith Degen, Michael Franke, and Gerhard Jäger. Optimal reasoning about referential expressions. 2012.

David DeVault and Matthew Stone. Managing ambiguities across utterances in dialogue. In *Proceedings of the 11th Workshop on the Semantics and Pragmatics of Dialogue (Decalog 2007)*, pages 49–56, 2007.

Hubert L. Dreyfus. *What computers can't do: A critique of artificial reason*. Harper & Row New York, 1972.

Jacob Eisenstein, James Clarke, Dan Goldwasser, and Dan Roth. Reading to learn: Constructing features from semantic abstracts. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 958–967, Singapore, August 2009. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/D/D09/D09-1100`.

Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. Every picture tells a story: Generating sentences from images. In *European Conference on Computer Vision (ECCV)*, pages 15–29. Springer, 2010.

Michele I. Feist. Space between languages. *Cognitive science*, 32(7):1177–1199, 2008.

Jerome Feldman and Srinivas Narayanan. Embodied meaning in a neural theory of language. *Brain and language*, 89(2):385–392, 2004.

Michael Fleischman and Deb Roy. Situated models of meaning for sports video retrieval. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*. Association for Computational Linguistics, 2007a.

Michael Fleischman and Deb Roy. Representing intentions in a cognitive model of language acquisition: Effects of phrase structure on situated verb learning. *Association for the Advancement of Artificial Intelligence (AAAI)*, 2007b.

Michael C. Frank and Noah D. Goodman. Predicting pragmatic reasoning in language games. *Science*, 336(6084):998–998, 2012.

Michael C. Frank, Noah D. Goodman, and Joshua B. Tenenbaum. Using speakers' referential intentions to model early cross-situational word learning. *Psychological Science*, 20(5): 578–585, 2009.

Gottlob Frege. *Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*. L. Nebert Halle, 1879.

Thomas Fuhr, Gudrun Socher, Christian Scheering, and Gerhard Sagerer. A three-dimensional spatial model for the interpretation of image data. In *International Joint Conferences on Artificial Intelligence (IJCAI-95) Workshop on Representation and Processing of Spatial Expressions*, volume 14, pages 93–102. Citeseer, 1995.

Klaus-Peter Gapp. *An empirically validated model for computing spatial relations*. Springer, 1995.

Claire Gardent. Generating minimal definite descriptions. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 96–103. Association for Computational Linguistics, 2002.

Ruifang Ge and Raymond J. Mooney. A statistical semantic parser that integrates syntax and semantics. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, pages 9–16. Association for Computational Linguistics, 2005.

Dave Golland, Percy Liang, and Dan Klein. A Game-Theoretic Approach to Generating Spatial Descriptions. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 410–419, Cambridge, MA, October 2010. Association for Computational Linguistics.

Peter Gorniak and Deb Roy. Grounded semantic composition for visual scenes. *Journal of Artificial Intelligence Research (JAIR)*, 21:429–470, 2004.

H. P. Grice. Syntax and Semantics; Logic and Conversation. 3:Speech Acts:41–58, 1975.

Sergio Guadarrama and David P. Pancho. Using soft constraints to interpret descriptions of shapes. In *Semantic Computing (ICSC), 2010 IEEE Fourth International Conference on*, pages 341–348. IEEE, 2010.

Sonal Gupta and Raymond J. Mooney. Using closed captions to train activity recognizers that improve video retrieval. In *Computer Vision and Pattern Recognition Workshops, 2009. CVPR Workshops 2009. IEEE Computer Society Conference on*. IEEE, 2009.

William G. Hayward and Michael J. Tarr. Spatial language and spatial representation. *Cognition*, 55(1):39 – 84, 1995. ISSN 0010-0277. doi: 10.1016/0010-0277(94)00643-Y. URL `http://www.sciencedirect.com/science/article/pii/001002779400643Y`.

Annette Herskovits. *Language and spatial cognition*. Cambridge University Press, 1987.

Geoffrey E. Hinton. Products of experts. In *Artificial Neural Networks, 1999. ICANN 99. Ninth International Conference on (Conf. Publ. No. 470)*, volume 1, pages 1–6. IET, 1999.

Kaijen Hsiao, Sachin Chitta, Matei Ciocarlie, and E. Gil Jones. Contact-reactive grasping of objects with partial shape information. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 1228–1235. IEEE, 2010.

Dominic Hyde. Sorites paradox. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Winter 2011 edition, 2011.

Naoto Iwahashi, Komei Sugiura, Ryo Taguchi, Takayuki Nagai, and Tadahiro Taniguchi. Robots that learn to communicate: A developmental approach to personally and physically situated human-robot conversations. In *AAAI Fall Symposia on Dialog with Robots*, 2010.

Gerhard Jäger. Game theory in semantics and pragmatics. *manuscript, University of Bielefeld*, 2008.

Yangqing Jia, Chang Huang, and Trevor Darrell. Beyond spatial pyramids: Receptive field learning for pooled image features. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012.

Rohit J. Kate and Raymond Mooney. Semi-supervised learning for semantic parsing using support vector machines. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*. Association for Computational Linguistics, 2007. URL `http://www.aclweb.org/anthology/N/N07/N07-2021`.

Rohit J. Kate and Raymond J. Mooney. Using string-kernels for learning semantic parsers. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 913–920. Association for Computational Linguistics, 2006.

Rohit J. Kate, Yuk Wah Wong, and Raymond J Mooney. Learning to transform natural to formal languages. In *Proceedings of the National Conference on Artificial Intelligence*, volume 20, page 1062. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005.

John D. Kelleher and Fintan J. Costello. Applying computational models of spatial prepositions to visually situated dialog. *Computational Linguistics*, 35(2):271–306, 2009.

John D. Kelleher, Geert-Jan M. Kruijff, and Fintan J. Costello. Proximity in context: An empirically grounded computational model of proximity for processing topological spatial expressions. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 745–752, Sydney, Australia, July 2006. Association for Computational Linguistics. doi: 10.3115/1220175.1220269. URL `http://www.aclweb.org/anthology/P06-1094`.

Joohyun Kim and Raymond Mooney. Generative alignment and semantic parsing for learning from ambiguous supervision. In *Coling 2010: Posters*. Coling 2010 Organizing Committee, 2010. URL `http://www.aclweb.org/anthology/C10-2062`.

Robert Kirchner. Phonological contrast and articulatory effort. *Segmental phonology in Optimality Theory. Constraints and Representations, edited by Linda Lombardi*, pages 79–117, 2001.

Thomas Kollar, Stefanie Tellex, Deb Roy, and Nicholas Roy. Toward understanding natural language directions. In *Proceedings of the 5th ACM/IEEE international conference on Human-robot interaction*, HRI '10, Piscataway, NJ, USA, 2010. IEEE Press. ISBN 978-1-4244-4893-7. URL `http://dl.acm.org/citation.cfm?id=1734454.1734553`.

Emiel Krahmer and Kees Van Deemter. Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218, 2012.

Emiel Krahmer, Sebastiaan van Erk, and André Verleg. Graph-based generation of referring expressions. *Computational Linguistics*, 29(1):53–72, 2003.

Benjamin Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119(1):191–233, 2000.

Girish Kulkarni, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C. Berg, and Tamara L. Berg. Baby talk: Understanding and generating simple image descriptions. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1601–1608. IEEE, 2011.

Tom Kwiatkowksi, Luke Zettlemoyer, Sharon Goldwater, and Mark Steedman. Inducing probabilistic CCG grammars from logical form with higher-order unification. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2010. URL `http://www.aclweb.org/anthology/D10-1119`.

George Lakoff and Mark Johnson. *Metaphors we live by*. University of Chicago press, 2008.

Barbara Landau and Ray Jackendoff. "what" and "where" in spatial language and spatial cognition. *Behavioral and Brain Sciences*, 16:217–238, 1993.

Douglas B. Lenat, Mayank Prakash, and Mary Shepherd. Cyc: Using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks. *AI magazine*, 6(4): 65, 1985.

Michael Levit and Deb Roy. Interpretation of spatial language in a map navigation task. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 37(3):667–679, 2007.

Percy Liang, Michael Jordan, and Dan Klein. Learning semantic correspondences with less supervision. In *Association for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, Singapore, 2009. Association for Computational Linguistics.

Percy Liang, Michael Jordan, and Dan Klein. Learning dependency-based compositional semantics. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 2011. URL `http://www.aclweb.org/anthology/P11-1060`.

Gordon D. Logan and Daniel D. Sadler. *Language and Space*, chapter A computational analysis of the apprehension of spatial relations. Bradford Books, 1996.

Wei Lu, Hwee Tou Ng, Wee Sun Lee, and Luke S. Zettlemoyer. A generative model for parsing natural language to meaning representations. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 783–792. Association for Computational Linguistics, 2008.

Matt MacMahon, Brian Stankiewicz, and Benjamin Kuipers. Walk the talk: Connecting language, knowledge, and action in route instructions. *Def*, 2(6):4, 2006.

Pascal Matsakis and Laurent Wendling. A new way to represent the relative position between areal objects. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(7): 634–643, 1999.

Cynthia Matuszek, Dieter Fox, and Karl Koscher. Following directions using statistical machine translation. In *Proceedings of the 5th ACM/IEEE international conference on Human-robot interaction*. IEEE Press, 2010.

Margaret Mitchell, Xufeng Han, Jesse Dodge, Alyssa Mensch, Amit Goyal, Alex Berg, Kota Yamaguchi, Tamara Berg, Karl Stratos, and Hal Daumé III. Midge: Generating image descriptions from computer vision detections. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 747–756. Association for Computational Linguistics, 2012.

Reinhard Moratz and Thora Tenbrink. Spatial reference in linguistic human-robot interaction: Iterative, empirically supported development of a model of projective relations. *Spatial Cognition and Computation*, 6(1):63–107, 2006.

Mikio Nakano, Naoto Iwahashi, Takayuki Nagai, Taisuke Sumii, Xiang Zuo, Ryo Taguchi, Takashi Nose, Akira Mizutani, Tomoaki Nakamura, Muhanmad Attamim, Hiromi Narimatsu, Kotaro Funakoshi, and Yuji Hasegawa. Grounding new words on the physical world in multi-domain human-robot dialogues. In *AAAI Fall Symposia on Dialog with Robots*, 2010.

Seungho Nam. *The semantics of locative prepositional phrases in English*. PhD thesis, University of California, Los Angeles, 1995.

John O'Keefe. The spatial prepositions in english, vector grammar, and the cognitive map theory. *Language and space*, pages 277–316, 1996.

Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. Im2text: Describing images using 1 million captioned photographs. In *Neural Information Processing Systems (NIPS)*, 2011.

Prashant Parikh. A game-theoretic account of implicature. In *Proceedings of the 4th conference on Theoretical aspects of reasoning about knowledge*, pages 85–94. Morgan Kaufmann Publishers Inc., 1992.

Slav Petrov and Dan Klein. Improved inference for unlexicalized parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*. Association for Computational Linguistics, 2007. URL `http://www.aclweb.org/anthology/N/N07/N07-1051`.

Steven T. Piantadosi, Noah D. Goodman, Benjamin A. Ellis, and Joshua B. Tenenbaum. A bayesian model of the acquisition of compositional semantics. In *Proceedings of the Thirtieth Annual Conference of the Cognitive Science Society*, pages 1620–1625, 2008.

Hoifung Poon and Pedro Domingos. Unsupervised semantic parsing. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2009. URL `http://www.aclweb.org/anthology/D/D09/D09-1001`.

Terry Regier and Laura A. Carlson. Grounding spatial language in perception: An empirical and computational investigation. *Journal of Experimental Psychology General*, 130(2): 273–298, 2001.

Ehud Reiter and Robert Dale. *Building natural language generation systems*. Cambridge university press, 2000.

Hannah Rohde, Scott Seyfarth, Brady Clark, Gerhard Jaeger, and Stefan Kaufmann. Communicating with cost-based implicature: a game-theoretic approach to ambiguity. In *The 16th Workshop on the Semantics and Pragmatics of Dialogue*, 2012.

Deb K. Roy. Learning visually grounded words and syntax for a scene description task. *Computer Speech & Language*, 16(3):353–385, 2002.

Bertrand Russell. On denoting. *Mind*, 14(56):479–493, 1905.

Jeffrey Mark Siskind. Grounding language in perception. *Artificial Intelligence Review*, 8 (5-6):371–391, 1994.

Marjorie Skubic, Dennis Perzanowski, Sam Blisard, Alan Schultz, William Adams, Magda Bugajska, and Derek Brock. Spatial language for human-robot dialogs. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 34(2):154–167, 2004.

Roy A. Sorensen. An argument for the vagueness of 'vague'. *Analysis*, 45(3):134–137, 1985.

Mark J. Steedman. Gapping as constituent coordination. *Linguistics and philosophy*, 13(2): 207–263, 1990.

Luc Steels and Paul Vogt. Grounding adaptive language games in robotic agents. In *European Conference on Artificial Life, Cambridge, MA*, 1997.

Charles Sutton and Andrew McCallum. An introduction to conditional random fields for relational learning. In Lise Getoor and Ben Taskar, editors, *Introduction to Statistical Relational Learning*. MIT Press, 2007.

Lappoon R. Tang and Raymond J. Mooney. Using multiple clause constructors in inductive logic programming for semantic parsing. In *European Conference on Machine Learning (ECML) 2001*, pages 466–477. Springer, 2001.

Stefanie Tellex. *Natural language and spatial reasoning*. PhD thesis, Massachusetts Institute of Technology, 2010.

Stefanie Tellex and Deb Roy. Grounding spatial prepositions for video search. In *Proceedings of the 2009 international conference on Multimodal interfaces*, pages 253–260. ACM, 2009.

Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew R Walter, Ashis Gopal Banerjee, Seth Teller, and Nicholas Roy. Understanding Natural Language Commands for Robotic Navigation and Mobile Manipulation. In *Proceedings Of The National Conference On Artificial Intelligence*, number Aaai, 2011a.

Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew R. Walter, Ashis Gopal Banerjee, Seth Teller, and Nicholas Roy. Approaching the symbol grounding problem with probabilistic graphical models. *AI Magazine*, 32(4), 2011b.

Sebastian Varges. Overgenerating referring expressions involving relations and booleans. In *Natural Language Generation*, pages 171–181. Springer, 2004.

Adam Vogel and Dan Jurafsky. Learning to follow navigational directions. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 806–814. Association for Computational Linguistics, 2010.

Joseph Weizenbaum. Elizaa computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45, 1966.

Deirdre Wilson and Dan Sperber. Relevance theory. *Handbook of pragmatics*, 2002.

Simon AJ Winder and Matthew Brown. Learning local image descriptors. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

Terry Winograd. Understanding natural language. *Cognitive Psychology*, 3(1):1–191, 1972.

Ludwig Wittgenstein. Philosophical investigations. blackwells, 1953.

Yuk Wah Wong and Raymond Mooney. Learning synchronous grammars for semantic parsing with lambda calculus. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Association for Computational Linguistics, 2007. URL `http://www.aclweb.org/anthology/P07-1121`.

Yuk Wah Wong and Raymond J. Mooney. Learning for semantic parsing with statistical machine translation. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 439–446. Association for Computational Linguistics, 2006.

K. M. Wurm, A. Hornung, M. Bennewitz, C. Stachniss, and W. Burgard. OctoMap: A probabilistic, flexible, and compact 3D map representation for robotic systems. In *Proc. of the ICRA 2010 Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation*, Anchorage, AK, USA, May 2010. URL `http://octomap.sf.net/`. Software available at `http://octomap.sf.net/`.

Atsushi Yamada, Toyoaki Nishida, and Shuji Doshita. Figuring out most plausible interpretation from spatial descriptions. In *Proceedings of the 12th conference on Computational linguistics - Volume 2*, COLING '88, pages 764–769, Stroudsburg, PA, USA, 1988. Association for Computational Linguistics. ISBN 963 8431 56 3. doi: 10.3115/991719.991792. URL `http://dx.doi.org/10.3115/991719.991792`.

Yezhou Yang, Ching Lik Teo, Hal Daumé III, and Yiannis Aloimonos. Corpus-guided sentence generation of natural images. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 444–454. Association for Computational Linguistics, 2011.

Chen Yu and Dana H. Ballard. On the integration of grounding language and learning objects. In *Proceedings of the National Conference on Artificial Intelligence*. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2004.

Lotfi Asker Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

John M. Zelle and Raymond J. Mooney. Learning to parse database queries using inductive logic programming. In *Proceedings of the National Conference on Artificial Intelligence*, 1996.

Hendrik Zender, Geert-Jan M. Kruijff, and Ivana Kruijff-Korbayová. Situated resolution and generation of spatial referring expressions for robotic assistants. In *Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI-09)*, pages 1604–1609, 2009.

Luke Zettlemoyer and Michael Collins. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. *Uncertainty in Artificial Intelligence (UAI)*, 5, 2005.

Luke Zettlemoyer and Michael Collins. Online learning of relaxed CCG grammars for parsing to logical form. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Association for Computational Linguistics, 2007. URL `http://www.aclweb.org/anthology/D/D07/D07-1071`.

C. Lawrence Zitnick and Devi Parikh. Bringing semantics into focus using visual abstraction. In *Computer Vision and Pattern Recognition. IEEE Computer Society Conference on*. IEEE, 2013.

Jordan Zlatev. *The Oxford handbook of cognitive linguistics*, chapter Spatial Semantics. Oxford University Press, USA, 2007.

Joost Zwarts and Yoad Winter. Vector space semantics: A model-theoretic analysis of locative prepositions. *Journal of logic, language and information*, 9(2):169–211, 2000.