

NORTHWESTERN UNIVERSITY

Using Analogy to Model Spatial Language Use and Multimodal Knowledge
Capture

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL
IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS

for the degree

DOCTOR OF PHILOSOPHY

Field of Computer Science

By

Kate Lockwood

EVANSTON, ILLINOIS

DECEMBER 2009

© Copyright by Kate Lockwood 2009
All Rights Reserved

ABSTRACT

Using Analogy to Model Spatial Language Use and Multimodal Knowledge Capture

Kate Lockwood

Language and knowledge capture are two skills that allow us to create structured representations of the physical world for reasoning and communication. Spatial prepositions are a form of specialized language that is used to relate two objects in space. In addition to communicating the static location of objects, spatial prepositions contain layers of information about the interactions and potential interactions between objects, agents, and their environments. While a large amount of information is encoded in spatial prepositions, the components of scene that contribute to their use are only a fraction of the possible features available. In order to learn the correct spatial prepositions categories in a language, a learner must figure out how to abstract the important concepts without being distracted by surface features.

When communicating about complex spatial relationships, a diagram can often, as the saying goes, “be worth a thousand words”. Diagrams can communicate complex spatial concepts concisely, which is why they are often used in educational materials. Information from text accompanying diagrams must be integrated with the spatial information from the diagram to create a cohesive understanding of the concepts being communicated. This process is called *multimodal knowledge capture*. It is multimodal because the information being captured is presented in two different modalities: text and diagrams. Often spatial language, in particular spatial prepositions, in the text provides clues about how to best integrate the different representations.

This dissertation addresses computational modeling of both learning spatial prepositions and multimodal knowledge capture. In particular, it examines the role that structure-mapping plays in both tasks. In spatial preposition use, sequential generalization over multiple scenes results in the

abstraction of core category concepts. In multimodal knowledge capture, structure mapping provides a framework for integrating structured representations from different modalities.

DEDICATION

This dissertation is dedicated to my grandmother, Maxine Tomkinson.

ACKNOWLEDGEMENTS

There are countless people, without whom, this dissertation never would have been finished. I can't possibly thank each of you enough, and I would need a second volume to list everyone's name. Even if you are not listed here, please know that your friendship and your support has meant a lot to me over the last six years.

First of all, I would like to thank my advisor Ken Forbus. Ken took me on as a lost, second-year student, and gave me the space and time to explore my academic interests. I would not be here today if it weren't for his mentoring and his unwavering confidence in my ability to do research – even when I personally doubted it. I will always be grateful for the amazing opportunities I had as a graduate student in QRG. I owe this dissertation (and my improved understanding of comma usage) to his patient guidance and support.

I would also like to thank my committee members Dedre Gentner and Ian Horswill for their advice and feedback over the years. To Bryan Pardo who was not just a committee member, but also a friend – thank you for taking time to talk me through my daily crises and for convincing me that graduate school would be (mostly) worth it.

To all of my lab mates, past and present in QRG – I don't know if a simple thank you could ever capture how much I appreciate your friendship over the years. I could always count on you to help work through a tricky idea, or to trip to the Oasis, depending on what the occasion warranted. I would especially like to thank Emmet Tomai, Matt Klenk, and Morteza Dehghani. I could not have made it through the last two years without our dissertation support group. I am very lucky to count you all as both colleagues and friends. I also owe a huge thank you to Jeff Usher, for his patient tutoring and infectious enthusiasm for LISP. And to Jenn Stedillie whose friendship and ability to cut through

University red-tape were both vital to my success as a graduate student. Thank you to all of my FREECS friends for helping me to laugh when I wanted to cry, quit graduate school, or both.

Whatever semblance of sanity I was able to maintain while working on this dissertation is owed to my amazing Chicago friends. I am grateful for your cheerleading, your understanding when I had to blow off plans for work, and most of all because you never asked when I would be done. My Frisbee friends from Chicago are truly among the best friends I have ever had, or could hope to have. You guys are amazing, and I miss you already. To Kirsten: thank you for being my partner in crime, even after you ended up with a broken thumb. To Marla and Sarah: thank you for always being there to get fries and cheer me up. To Regina and Waggs: I know we don't speak often, but you guys mean more than me than you can ever know – you have both forgiven more than most people would and have had my back even when I was being ridiculous. Regina, you will be proud to know I did not once use a ruler-line while preparing this dissertation.

My parents, Joyce and Alan Lockwood, have encouraged my educational pursuits from the beginning and have been amazingly supportive of my perpetual student-hood. Thank you, Dad, for teaching me to write my first program on the Commodore 64 – I bet you didn't know it would lead to all this! Thank you for teaching me that something worth doing (e.g., calculus) is worth doing right, even if I didn't appreciate the message at the time. Thank you, Mom, for reminding me to slow down and pursue outside interests and for everything you have done for me over the years. Thank you for your unwavering support for everything I have ever tried (no matter how misguided). Thank you for always giving your honest advice, and for waiting until I asked for it. I probably should have asked more often. I hope that you both are proud; this accomplishment is yours as much as mine.

To my brother, John Lockwood, thank you for never letting me take myself too seriously. I value the friendship that we have developed immensely and I really appreciate your ability to put things into

perspective. It may have taken 30 years or so, but now I can say that I think Mom and Dad make the right call by not taking our pajamas off of you and taking you back to the hospital.

Last, but certainly not least, thank you to my husband, Jeremy Gottlieb who knew what he was getting into but went ahead and married a graduate student anyway. I am so lucky that we found each other (even if we disagree about exactly how that happened). I can't image spending my life with anyone else. Thank you for your patience, your understanding and your encouragement. Now that the dissertation is finally behind us, I can't wait for our next adventure.

TABLE OF CONTENTS

1	Introduction	16
1.1	Modeling the Learning and Use of Spatial Language.....	18
1.2	Multimodal Knowledge Capture.....	20
1.3	Claims and Contributions.....	22
1.4	Dissertation Organization	23
2	Theoretical Background	24
2.1	Introduction	24
2.2	Acquisition and Use of Spatial Prepositions	25
2.2.1	Spatial Prepositions: The Basics	25
2.2.2	How Humans use Spatial Language/Implication for Computation.....	27
2.3	Multimodal Knowledge Capture.....	34
3	Systems Background	41
3.1	Introduction	41
3.2	Large Common Sense Knowledge Base and the FIRE Reasoning Engine.....	42
3.2.1	Common Sense Knowledge Base	42
3.2.2	The FIRE Reasoning Engine	43
3.3	Analogy and Similarity: SME, SEQL and MAC/FAC.....	44
3.3.1	SME	44
3.3.2	SEQL	45
3.3.3	MAC/FAC.....	46
3.4	EA NLU	47
3.5	CogSketch.....	48
3.6	Discussion.....	52
4	Spatial Preposition Experiments	53
4.1	Introduction	53
4.2	Problem Description	54
4.3	SpaceCase Model of Spatial Preposition Use	55
4.3.1	SpaceCase Experiment 1: Labeling	56
4.3.2	SpaceCase Experiment 2: The effect of spatial language on retrieval.....	66

	10
4.4 Geometric Shapes Experiments	69
4.4.1 Introduction	69
4.4.2 Geometric Shapes Experiment 1.....	70
4.4.3 Geometric Shapes Experiment 2.....	76
4.5 Cross Linguistic Experiments: Containment-Support relations in English and in Dutch.....	81
4.5.1 Introduction	81
4.5.2 Cross-linguistic Experiment.....	82
4.6 Related Work	93
4.7.4.7 General Discussion	96
5 Multimodal Knowledge Capture	99
5.1 Introduction	99
5.2 Materials	100
5.2.1 Text	101
5.2.2 Questions	102
5.3 The MMKCap Model	103
5.3.1 Overview	103
5.3.2 Selecting relevant words and images (steps 1 and 2).....	104
5.3.3 Organizing selected words (step 3).....	105
5.3.4 Organizing Selected Pictures (Step 4)	109
5.3.5 Integration (Step 5).....	111
5.4 Evaluation	114
5.4.1 Question Types and Answer Strategies	115
5.5 Results.....	122
5.6 Diagram Understanding.....	127
5.6.1 Problem Description: Conceptual Segmentation	127
5.6.2 Preliminary System and Results.....	130
5.7 Related Work	135
6 Conclusions and Future Work.....	140
6.1 Discussion.....	140
6.2 Future Work	142
6.2.1 Future Work in Spatial Prepositions	142

6.2.2	Future Work in Conceptual Segmentation of Diagrams	143
6.2.3	Future Work in Multimodal Knowledge Capture.....	143
6.3	Conclusion.....	146
7	Works Cited.....	147
8	Appendices.....	156
8.1	Appendix A: Stimuli for SpaceCase Experiment #2: Retrieval	157
8.2	Appendix B: Stimuli for Geometric Experiment 1.....	159
8.3	Appendix C: Facts filtered from the sketch cases.....	161
8.4	Appendix D: Generalizations created in Simple Geometric Experiment 1	162
8.5	Appendix E: Generalizations created by Geometric Shapes Experiment 2	164
8.6	Appendix F: Simple Geometric Experiment 2 Stimuli	166
8.7	Appendix G: Stimuli for Experiment # Learning Spatial Prepositions in English and in Dutch	167
8.8	Appendix H: Conceptual labels used in the Cross-Linguistic Experiment.....	169
8.9	Appendix I: Generalizations created for English <i>IN</i> and <i>ON</i> in the Cross-Linguistic Experiment when no test case is excluded (includes all training cases).....	170
8.10	Appendix J: Diagrams from <i>Basic Machines</i> Chapter 1.....	173
8.11	Appendix K: Homework questions for Chapter 1 in <i>Basic Machines</i>	178
8.12	Appendix L: Knowledge Added to the Knowledge Base to Facilitate Knowledge Capture	182
8.13	Appendix M: Filter used to extract bookkeeping information from sketch cases.....	187

LIST OF FIGURES

Figure 1. Two examples of multimodal information sources highlighting the importance of spatial language	17
Figure 2 Stimuli from human subjects studies of spatial preposition use.....	18
Figure 3 An example of multimodal input from the physics textbook Basic Machines consisting of both text and an accompanying diagram.	21
Figure 4. Preposition Classifications (Coventry and Garrod, 2004)	27
Figure 5. Transitivity for <i>on</i> in different situations	28
Figure 6. Example of a spatial template for <i>above</i> ,	30
Figure 7. Scenes used by Coventry, Prat-Sala and Richards (2001).....	31
Figure 8. Example stimuli from Coventry and Mather (2002)	32
Figure 9. Support and containment prepositions in various languages.	33
Figure 10. The five steps in Mayer’s multimedia learning theory	36
Figure 11. An example of multimedia learning in action.....	38
Figure 12. Two text/diagram pairs showing different uses of diagrams in text.....	40
Figure 13	49
Figure 14	51
Figure 15. Example stimuli from the original study (left) and the sketched equivalent (right).....	59
Figure 16. SpaceCase model	63
Figure 17. Example of 2x2 sensitivity analysis run on the SpaceCase likelihood values	65
Figure 18. Example Stimulus from the original Feist and Gentner study	67
Figure 20. Examples of the stimuli that served as inspiration for the inputs for Geometric Shapes Experiment 1. Figure a taken from (Regier & Carlson, 2001) b-e from (Regier, 1995)......	71
Figure 19. Examples of the sketched inputs for Geometric Shapes Experiment 1.....	71
Figure 21. The generalization created for the preposition <i>in</i>	73
Figure 22. Experimental design for Geometric Shapes Experiment 1	74
Figure 23. Examples of the ambiguous stimuli used in Geometric Shapes Experiment 2, highlighting the types of variations included	78
Figure 24. An example of one of the original examples of <i>on</i> and one of the <i>complex</i> examples.....	79
Figure 25. The new generalization for <i>on</i> after the complex sketches have been added.....	80
Figure 26. Two examples of <i>ann</i> drawn to highlight the type of connection.....	86

Figure 27. Experimental setup for the Cross-linguistic experiments.....	87
Figure 28. One generalization created for in	90
Figure 29. One generalization created for English <i>on</i> and the sketches that were the generalized cases	90
Figure 30. A worked example problem from the text	101
Figure 31. The diagram from Basic Machines associated with the worked example in Figure 30.....	102
Figure 32. An overview of the steps of the MMKCap model.....	103
Figure 33. Example of a multi-part diagram.	105
Figure 34. An example of a paragraph from the text and its associated diagram.....	107
Figure 35. The discourse case for chunk 2 of text in Figure 34.....	108
Figure 36. Example of the EA NLU manual disambiguation interface.	108
Figure 37. Several facts from the diagram case created for the diagram in Figure 34.....	111
Figure 38. An example discourse case showing how <code>sketchForDiscourse</code> facts are created.....	113
Figure 39. Candidate inferences from the integration of the discourse and diagram cases from the example in Figure 34	114
Figure 40. Example of a True/False question.....	116
Figure 41. Example of a simple query question.....	116
Figure 42. Example of a diagram-concept question	118
Figure 43. Retrieved example of a second class lever.	118
Figure 44. Example of a Diagram-Measurement Question.	120
Figure 45. Example of the sketch from the problem on the left and the retrieved best match on the right.	120
Figure 46. Example of an algebraic question.....	121
Figure 47. Example of an Algebraic w/Diagram Question	122
Figure 48. Example of the failed diagram-concept question.....	124
Figure 49. Diagram-measurement question that was answered incorrectly (left) and the known example retrieved to solve it (right).	125
Figure 50. Example of an algebra+diagram problem that is answered incorrectly.....	126
Figure 51. An example of a diagram that is easily segmented by humans but poses a problem for AI systems.	127
Figure 52. CogSketch sketch of a diagram of a tank of water..	128
Figure 53. Two examples of hard to segment sketched diagrams	129

Figure 54. Steps in conceptual segmentation algorithm	130
Figure 55. Decision tree of possible segmentation options for entities in diagrams.	131
Figure 56. Results from running conceptual segmentation.....	133
Figure 57. Sketched diagram of the solar system showing conceptual segmentation results for both an orbit and a solid planet.....	134
Figure 58. Results of segmentation algorithm with the query “water in tank1”	135

LIST OF TABLES

Table 1. The English Spatial Prepositions from (Herskovits, 1998).....	26
Table 2. Variations of scene factors taken from Feist and Gentner (2003).	58
Table 3. Rules for determining the label for ground function	59
Table 4. SpaceCase rules along with the preposition they support and their likelihood values	64
Table 5. Average applicability of the appropriate preposition (<i>in</i> or <i>on</i> depending on the stimulus) for each of the stimuli in the experiment as determined by SpaceCase	68
Table 6. Summary of the perceptual relationships that form the content of the generalizations created in Simple Geometry Experiment 1 and the categories in which they appear	76
Table 7. The containment and support spatial prepositions in English and Dutch	83
Table 8. The original stimuli from the Gentner and Bowerman experiment.....	84
Table 9. Results for both English and Dutch. All are significant, except for English <i>in</i>	88
Table 10. Number of generalizations and exemplars created within each context	88
Table 11. Summary of evaluation results.....	122

1 INTRODUCTION

Language and spatial reasoning are two of the skills that give people the ability to quickly and easily assimilate new knowledge and pass it along to others; both play key roles in multimodal knowledge capture and in spatial language use. Multimodal knowledge capture is the process of taking a source of information that contains multiple modalities (for example, text and diagrams) and converting it into the structured knowledge needed for later tasks. Spatial language, in particular spatial prepositions, has developed to enable people to communicate clearly about the locations of objects in the world. Often these processes happen so effortlessly that we do not stop to consider the mechanics behind them – nobody stops to think “how is it that I am able to convert this newspaper and its graphics into usable knowledge?” Likewise, one rarely stops to ponder why it is that they consider coffee to be *in* their cup.

While these types of processes are second-nature to people, they are quite hard for AI systems to accomplish. And although they seem like two unrelated processes, spatial language and capturing multimodal knowledge share some common structure. Both rely heavily on pre-existing world knowledge and spatial perception skills. We claim that aspects of both can also be modeled using the structure-mapping theory (Gentner, 1983) of analogy and similarity. Specifically, learning spatial language categories can be cast as a problem of progressively abstracting the salient common structure from labeled instances of a spatial relation. Similarly, multimodal knowledge capture requires mapping between representations built from each of the modalities. These are the main claims of this dissertation:

1. Sequential generalization can be used to model the learning of spatial prepositions, taking into account both functional and geometric features of a scene. In addition, sequential

generalization can learn spatial preposition categories using far fewer training trials than existing models.

2. Structure mapping can be used to model the integration of multi-modal knowledge sources in a domain-general fashion without relying on predefined, domain-specific conventions.

These claims sit within a larger theoretical framework that applies structure-mapping principles to spatial reasoning, spatial language, and perception tasks. The physical world contains a large amount of naturally occurring structure and we rely on that structure, which we encode and represent via structured, relational representation, to navigate in the world, to communicate about our place in that world, and to reason about the world. Our language and visual representation conventions have also developed to be highly structured. Structure-mapping provides a framework for comparing and abstracting the structural features of both visual and natural language stimuli. Aligning common structure allows us to attend to the deeper, spatial and functional commonalities between representations regardless of the modality or differing surface features.

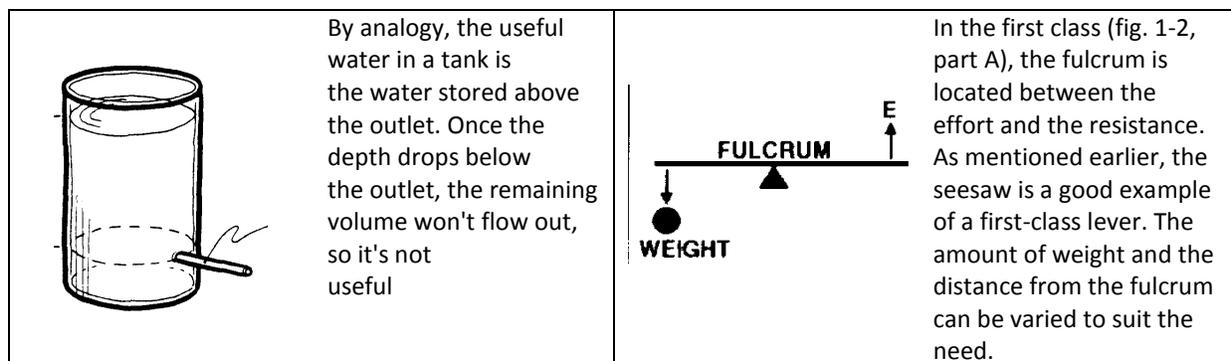


Figure 1. Two examples of multimodal information sources highlighting the importance of spatial language

Furthermore, spatial language plays an important role in multimodal knowledge capture. The two examples of multimodal information in Figure 1 show how spatial language can give insights into how to interpret diagrams. In the example on the left, the prepositions explain how the steps of a

process occur in different areas of the diagram. In the example on the right, the prepositions highlight the important spatial relationships in the diagram. This dissertation presents a series of experiments on learning and using spatial prepositions and an evaluation of a multimodal knowledge capture system. While these two efforts are, for the moment, separate entities, the conclusion lays out a plan for how they might be combined in future work.

1.1 MODELING THE LEARNING AND USE OF SPATIAL LANGUAGE

The problem of modeling spatial language use is decoding how people map from relationships in the world to a small, closed-class set of words (the spatial prepositions). There have been different approaches to this question ranging from minimal specification to full specification. Minimal specification approaches (e.g. Miller & Johnson-Laird, 1976) take the stance that there are a small

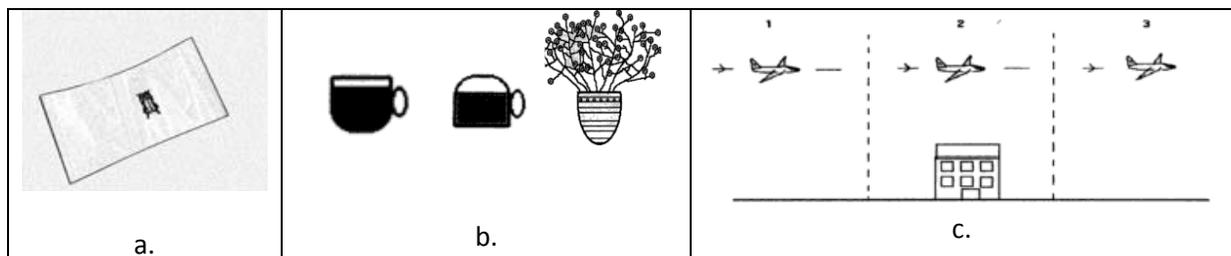


Figure 2 Stimuli from human subjects studies of spatial preposition use.

number of template-like structures that indicate how to carve Euclidean space into different regions associated with each preposition. Under minimal specification, spatial language use comes down to learning how to manipulate the templates to fit different situations. At the other end of the spectrum, full specification approaches (e.g. Herskovits, 1985, 1986; Brugman & Lakoff, 1988; Lakoff, 1987) call for creating a large number of templates, one for each possible meaning or use of a preposition. Under full specification, using a preposition comes down to choosing the correct template/sense.

More recent work in cognitive psychology has moved beyond looking at the placement of objects in space to considering the other factors that influence spatial preposition usage. Consider Figure 2 which highlights stimuli from experiments trying to discover how different aspects of a scene influence spatial prepositions use. Part a is an example stimulus from a study by Feist and Gentner (2003) showing that ground label, ground curvature, ground animacy, figure animacy and label of the ground (e.g. “bowl” vs “dish”) all impacted subjects’ judgments of *in* and *on*. Part b shows an example from Coventry (1998) showing an acceptable example of “the coffee is *in* the cup” along with an unacceptable example of the same situation and an acceptable example of “the flowers are *in* the vase”. These three examples are meant to demonstrate the importance of the containment role of the preposition *in*. In the two acceptable uses of *in* the ground object is functionally containing the figure object. In the unacceptable example of the coffee being *in* the cup, it is impossible that the cup is actually fulfilling its role of containing the coffee despite the fact that the objects in that instance are in the same general spatial arrangement as the flowers *in* the vase. Part c is an example stimulus from Coventry and Mather (2002) who demonstrated the role of naïve physics in spatial preposition use, and provided further evidence that some prepositions such as *over* are much more sensitive to function information than others such as *above*. In addition to studies examining single languages, a number of cross-linguistic experiments have investigated the varying ways that different languages carve up the space of spatial relations (e.g. McDonough, Choi and Mandler, 2003; Gentner and Bowerman, 2009).

The aspects of a visual scene that are listed above are just a sampling of the factors that have been shown to impact spatial language use. The same diversity that makes spatial language an interesting topic of study for cognitive psychologists has drawn in the computer science community. There have been a variety of attempts to model subsets of spatial prepositions, employing multiple

techniques from machine learning and artificial intelligence (for example Regier, 1995; Cangelosi *et al*, 2005).

1.2 MULTIMODAL KNOWLEDGE CAPTURE

When humans solve problems, they rely on a variety of previous knowledge – everything from commonsense knowledge about objects and actions in the world to advanced, domain specific knowledge. Artificial Intelligence systems designed to do the same tasks necessarily need access to the same kinds of information. This problem is solved by providing such systems with knowledge bases containing the facts and axioms necessary for a given task or set of tasks. The construction of knowledge bases is currently done primarily by hand. This process is time consuming, expensive, and can lead to knowledge that is overly tailored to a specific domain or problem. For example the Cyc knowledge base (Lenant, 1995), the largest effort at a common sense knowledge base, has been in development since 1984 at an estimated cost of at least \$50 million and over 600 man-years of effort (2002). Today's Cyc has around 3 million rules of thumb plus around 300,000 terms or concepts, but still falls short of human-like common sense knowledge. The HALO project is attempting to build a system that can answer AP test questions in Biology, Chemistry and Physics. Their knowledge base is hand built by subject matter experts, at a reported time of 22 minutes per concept (Chaudhri *et al*, 2007). Projects such as these, while effective, are clearly not ideal.

To get around the problem of hand-constructed knowledge bases, the learning by reading (LbR) community has been creating systems that can automatically construct structured knowledge from natural language sources like books, newspapers, and the web. While much progress has been made in LbR, one area that remains largely underexplored is knowledge capture from multimodal information sources. Multimodal sources are those where the information is presented in more than one modality – such as narration and animation or text and diagrams. For example, HALO discards the diagrams from

their sources and the MOBIUS (Barker *et al*, 2007) project also does not handle diagrams despite operating in a domain that includes highly structured spatial information (the structure of the human heart). This is a particularly interesting combination since it is pervasive in educational materials and often the information in the diagram is not reproduced in the text. Figure 3 below shows an example of the kind of multimodal information that we are interested in capturing. You can see that both the text and the diagram are necessary for a full understanding of the concepts being communicated.

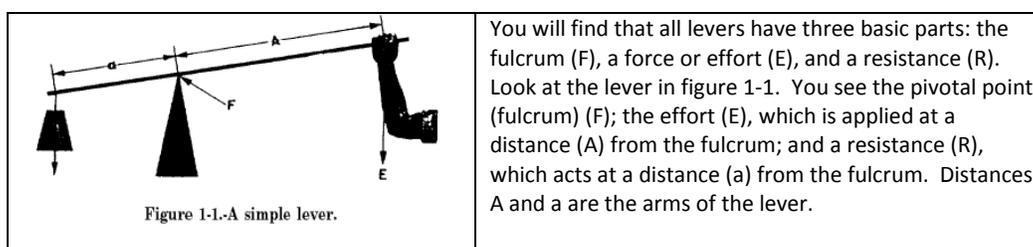


Figure 3 An example of multimodal input from the physics textbook *Basic Machines* consisting of both text and an accompanying diagram.

There is a lot of research on how people learn from multimodal information sources, and under many circumstances, learning is better from multimodal sources than from single modality sources (for an overview see Mayer, 2001). So, it makes sense that the LbR community should be interested in exploiting these types of information sources as well. Indeed, many traditional sources of instructional materials, such as textbooks, contain multiple modalities of information. Studies examining how and why people learn from multimodal sources of information appear to indicate that people engage cognitively with the material, at times going back and forth between the text and the diagram in an attempt to create a coherent mental representation of the information being presented (Hegarty & Just 1993; Mayer, 2001). This process of integration appears to involve a mapping between the description in the text and the spatial objects in the diagram which lends itself well to modeling via analogy.

1.3 CLAIMS AND CONTRIBUTIONS

This dissertation contains two sets of experiments. The first set examines the formation and use of spatial prepositions and the second involves the creation and evaluation of a multimodal knowledge capture system.

The first set of experiments examines the use of spatial prepositions. As spatial language use has been a topic of much inspection by cognitive psychologists, there have also been many attempts to computationally model subsets of spatial prepositions. The work in this dissertation takes a different approach than existing models by using analogical generalization to model the formation of spatial categories. This approach allows for a more knowledge-rich approach than others and also can demonstrate learning in a smaller, more plausible number of inputs and trials. There are two sets of experiments using this technique: one looking at five English prepositions (*in, on, above, below, and left*) and another modeling the support-containment relations in English and in Dutch. Two additional experiments show how the kind of output produced by generalization might be used as part of a system that labels prepositions in novel scenes and can account for some results showing the effect of spatial language on memory.

The second set of experiments demonstrates the use of analogy to model integration during multimodal knowledge capture. The multimodal knowledge capture system built uses a combination of sketched diagrams and simplified English to represent the input source. The sketches are turned into structured representations using CogSketch and the text is processed using EA NLU (both of which are described in Chapter 3). Then, integration of the two representations is done using the SME model of analogy and similarity. This approach to multimodal knowledge capture is much more flexible and domain-independent than many other similar systems. The performance of the system is evaluate on

the contents of a simple physics textbook (*Basic Machines*) by evaluating the system's ability to answer the publisher-provided homework assignments.

1.4 DISSERTATION ORGANIZATION

Chapter 2 provides the theoretical background for both the spatial language and knowledge capture work. This background is drawn from the literature in cognitive psychology and learning sciences.

Chapter 3 describes the existing systems that were used in different parts of this work. Included are the large-scale knowledge base and reasoning engine that underlie all of the systems. Also included are three related models of analogy: the SME model of analogy and similarity, the SEQL model of analogical generalization, and the MAC/FAC model of similarity-based retrieval. EA NLU and CogSketch which provide a means of input for the experiments in this dissertation are also introduced.

Chapter 4 describes three experiments modeling the formation and use of spatial language categories. Related work is also discussed.

Chapter 5 introduces a multimodal knowledge capture system and summarizes experiments that show its performance over a simple physics textbook. The results of a system evaluation based on its ability to answer publisher-provided homework questions are provided. Related work on multimodal knowledge capture, diagram understanding, and learning by reading is discussed.

Chapter 6 returns to the claims of this thesis, summarizes the results and discusses some important future directions including how the two themes of the dissertation may be combined in future work.

2 THEORETICAL BACKGROUND

2.1 INTRODUCTION

The experiments in this thesis model processes that occur at the intersection of language and space. Spatial prepositions are used to communicate the spatial arrangements of objects in a quick and compact format. Learning to use them correctly requires understanding which features of a spatial scene need to be attended to in order to form the agreed upon set of categories in a given language. Multimodal knowledge capture requires being able to integrate information from language (text) and spatial information from a diagram. Conceptual segmentation of diagrams involves being able to extract the correct region or edge from a diagram based on a language cue. This requires not only attending to both modalities, but also understanding which features of an object play a role in how its spatial extent is determined.

To computationally model cognitive processes involving language and space, it is important to first understand current theories of how these processes occur in human subjects. First of all, human subjects studies provide a ready supply of carefully crafted and normed stimuli as fodder for cognitive modeling. Using these stimuli as input into cognitive models reduces the extent to which inputs can be tailored to suit a given model's strengths. More importantly, by tying cognitive modeling closely to the underlying psychology, the two fields form a mutually beneficial feedback cycle. Psychology benefits by having another avenue to test their findings and to help identify potential gaps. AI benefits by creating systems that perform more like their human users and are therefore more intuitive to interact with.

This chapter first examines cognitive psychological theories of spatial preposition use. While some historical context is provided, the focus is on current theories which explore the role of functional information about objects as opposed to treating spatial relationships as purely geometric. The results

discussed in the spatial prepositions section provide the basis for the experiments in Chapter 4. Next, the literature on multimodal knowledge capture in humans is discussed, in particular the multimedia learning theory of Richard Mayer, on which the MMKCap model described in Chapter 5 is based.

2.2 ACQUISITION AND USE OF SPATIAL PREPOSITIONS

2.2.1 SPATIAL PREPOSITIONS: THE BASICS

Spatial prepositions describe the relation of one object (called the *located object*, or *figure*) with respect to another object (called the *reference object*, or *ground*). For example, in the sentence “the cup is *on* the table” the figure is the cup, the ground is the table, and *on* is the spatial preposition describing the relationship between the two. In English, spatial prepositions form a closed-class of words, and there are relatively few of them when compared with the large number of words in other syntactic categories (e.g., around 10,000 count nouns in the standard lexicon) (Landau & Jackendoff, 1993). Table 1 below enumerates the common spatial prepositions (leaving out domain-specific prepositions such as *aft* or *starboard*). Despite their relatively small number, many prepositions have a wide range of syntactic, semantic, and even idiomatic interpretations. Figure 4 (from Coventry and Garrod, 2004) shows a classification hierarchy for prepositions.

Even within the limited scope of space, the assignment of spatial prepositions to visual arrangements of objects is a complex cognitive process. At first glance, this problem appears to boil down to simply mapping linguistic labels to arrangements of objects in geometric space. However, there is no one-to-one mapping between language and the external world in the case of spatial prepositions, and a variety of non-geometric features have been found to influence spatial preposition use. Like many other cognitive tasks, coming up with a neat discretization for spatial language has proven to be quite difficult. One reason for this is the polysemy of spatial prepositions. Each

preposition can be correctly used in a variety of situations with slightly different meanings. For example, consider the following uses of the preposition *on*:

- (A) The mug is *on* the table.
- (B) Bob hung the clothes *on* the line.
- (C) Put your coat *on* before you go outside.
- (D) Sally got *on* the bus.

Figuring out how humans map from the complexities of the world to the small set of spatial prepositions is a key task for cognitive psychologists and computer modelers alike.

Table 1. The English Spatial Prepositions from (Herskovits, 1998)

Primarily Location	Primarily Motion
at/on/in	across
upon	along
against	to/from
inside/outside	around
within/without	away from
near/(far from)	toward
next	toward
beside	up/down to
by	up/down
between	into/(out of)
beyond	onto/off
opposite	out
amid	through
among	via
throughout	about
above/below	ahead of
under/over	past
beneath	
underneath	
alongside	
on top/bottom of	
on the top/bottom of	
behind	
in front/back of	
left/right of	
at/on/to the left/right/front/back of	
at/on/to the left/right side of	
north/east/west/south of	
to the east/north/south/west of	
on the east/north/south/west side of	

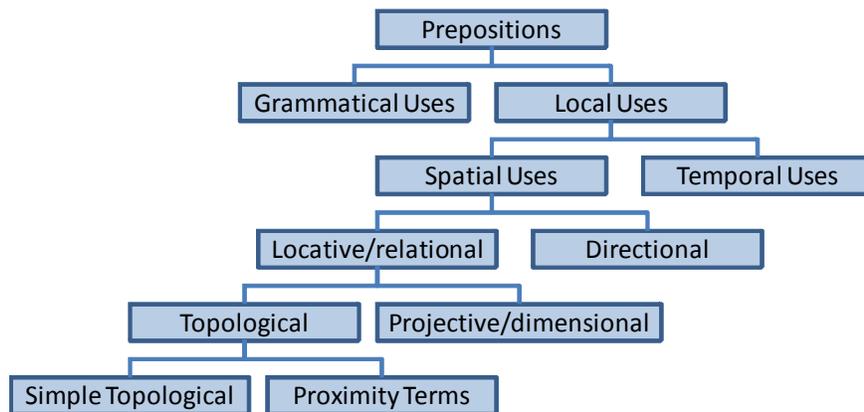


Figure 4. Preposition Classifications (Coventry and Garrod, 2004)

2.2.2 HOW HUMANS USE SPATIAL LANGUAGE/IMPLICATION FOR COMPUTATION

The use of language to describe space is an area with a long history in psychological research. Early theories of spatial preposition use claimed that people assigned spatial prepositions based solely on the geometry of a scene. They focused on developing minimally specified definitions for spatial prepositions. For example, Cooper (1968) suggested the following definition for “in”;

***IN:** X in Y: X is located internal to Y with the constraint that X is smaller than Y – where X is the located object and Y is the reference object.*

While this definition works for a number of cases where in is appropriate, it is relatively easy to find a counter-example, e.g., a bouquet of flowers in a vase. As counter-examples were found for existing definitions, others would be suggested. Here is another suggested definition for *in* (Miller & Johnson-Laird, 1976) which covers the case of the flowers in the vase:

IN(X, Y): A referent X is “in” a relatum Y if:

- (i) [PART(X, Z) & INCL (Z, Y)]

but again, counter-examples are relatively easy to find. Such definitional approaches to spatial prepositions showed two major problems:

- (1) Overgeneration: the generation of examples that should fit based on the definition, but do not actually work.
- (2) Situations where the definition does not fit, but the preposition is appropriate (like the flowers and vase example given above).

Another problem with definition based (also called minimal specification) accounts was explaining the constraints on transitivity. For example, consider figure 3a below. It is reasonable to say that the yellow atlas is on the table, despite the stack of books between the atlas and the table. However, in figure 3b, it is far less natural to say that the lid is on the table despite the fact that visually the situations are very similar.



Figure 5. Transitivity for *on* in different situations

All minimal specification accounts start from the assumption that there is some core definition for a preposition and that there were guidelines describing how that definition could be stretched to cover all of the possible uses of a given preposition. As it became clear that counter-examples could be found for any definition-based account of spatial preposition use, psychologists and linguists began to look at other influences. Searle (1979) discussed the importance of background conditions. One

example he used was that the phrase “the cat is on the mat” presupposes the background condition of gravity.

Another approach to spatial preposition definition was the exhaustive listing of all possible situations in which it was appropriate. Methods fitting this approach are often referred to as *full specification*. Annette Herskovits (1986) proposed a theory based on *use cases*. These use cases are normal situation types which Herskovits claims can be stretched to cover most cases of spatial preposition use. Several *near principles* are defined as the methods with which to stretch the use cases. The near principles are: salience, relevance, tolerance, and typicality.

Brugman (1988; 1981, as reported in Coventry & Garrod, 2004) listed over 100 kinds of uses for the preposition *over*. Approaches such as those of Herskovits and Brugman attempt to fully specify the cases in which a particular preposition is applicable. One problem with this approach is that there must be some mechanism to index and to select from between the different cases. Also, they rely on strategies like Herskovits’s near principles to fit many situations. There must also be a mechanism for selecting the near principle to apply and which case to modify. Another problem is that many of the cases depend on primitives like “higher than” – these primitives must be defined and recognized as well.

A similar approach is that of “spatial templates”. Logan and Sadler (1996) advocated a method of spatial templates centered on the reference object. The spatial templates for different prepositions would be evaluated for “goodness of fit”. An example of a template for *above* is shown in Figure 6. This approach faces the same problems as fully-specified accounts – how to select the correct template and how to find the best fit.

A	A	A	G	A	A	A
A	A	A	G	A	A	A
A	A	A	G	A	A	A
B	B	B	■	B	B	B
B	B	B	B	B	B	B
B	B	B	B	B	B	B
B	B	B	B	B	B	B

Figure 6. Example of a spatial template for *above*, where G = good region, A = acceptable region and B = bad region. Taken from Coventry and Garrod (2004) who adapted it from Carlson-Radvansky and Logan (1997)

More recent work in psychology has focused on the different features of a scene that influence spatial preposition use. In addition to geometry, many other factors have been shown to impact the use of spatial prepositions. One important factor is the functional relationship between the figure and the ground. Much of the recent work on “*in-ness*” and “*on-ness*” has focused on the roles that containment and support play. Whether the ground object is traditionally considered to be a container (e.g. Feist & Gentner, 1998), alternative sources of control (e.g. Coventry, 1999; Garrod, et. al., 1999) and liquid in the container (Coventry et al., 1994) can all influence how well an *in* relationship describes a visual scene.

Other work has highlighted the role that functional relationships play in other prepositions. Carlson-Radvansky and Radvansky (1996) show that *in front of* is more likely to be used when there is a functional relationship between the figure and ground (“the postman is *in front of* the mailbox”) as opposed to when there is no functional relationship (“the postman is *near* the birdhouse”). Additionally, the functional relationship must be able to be fulfilled in the scene, in the postman/mailbox example,

near is used instead of *in front of* if the mailbox opening is facing away from the postman. The fulfillment of functional roles has also been shown to influence the use of *over*. Coventry, Prat-Sala, and Richards (2001) showed that whether or not an umbrella was protecting a person from rain was key to whether or not it was described as *over* the person. This effect was extended to objects that don't normally function as protection (such as a briefcase held over a person to block the rain). Carlson-Radvansky et al. (1999) showed that people were more likely to describe a coin as *over* a piggy bank if it was lined up with the slot as opposed to lined up over the center of mass of the bank. When placing objects into given relationships, subjects are biased towards functionally related parts of the objects, for example, when placing a tube of toothpaste over a toothbrush, subjects are biased towards the bristles. This bias is lessened if the tube of toothpaste is replaced with a similarly-shaped tube of paint. Features of objects can also influence acceptability judgments, especially for objects where different features have different functional roles. Adding cartoon eyes to an object has a significant impact on acceptability judgments for *in front of* (Landau, 1996).

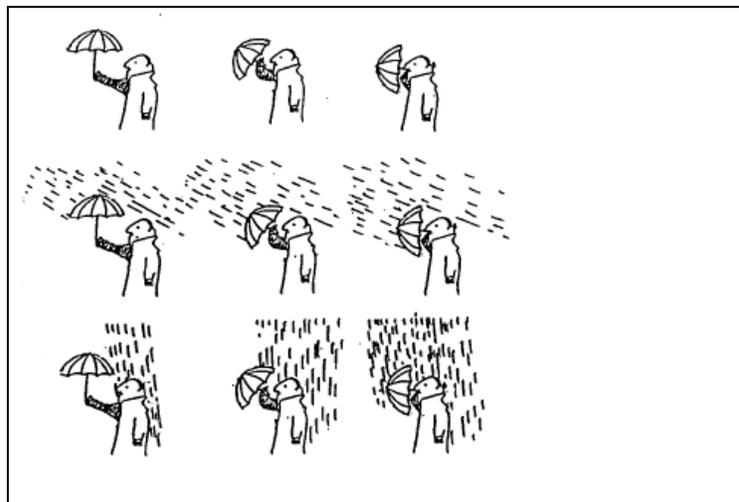


Figure 7. Scenes used by Coventry, Prat-Sala and Richards (2001)

Coventry and Mather (2002) demonstrated that *over* differs from *above* in that *over* is extremely sensitive to object-specific knowledge and functional relations between objects, including knowledge of the naïve physics between objects. In one study, subjects were presented with images of a plane and a building and were asked where the plane should be for the expression “the plane is *over* the building” to hold, either without a context, or within the context of the plane dropping a bomb on the building. With the context the ratings corresponded significantly with the location where the subjects thought the bomb should be dropped in order to hit the building as opposed to a canonical *above* relationship.

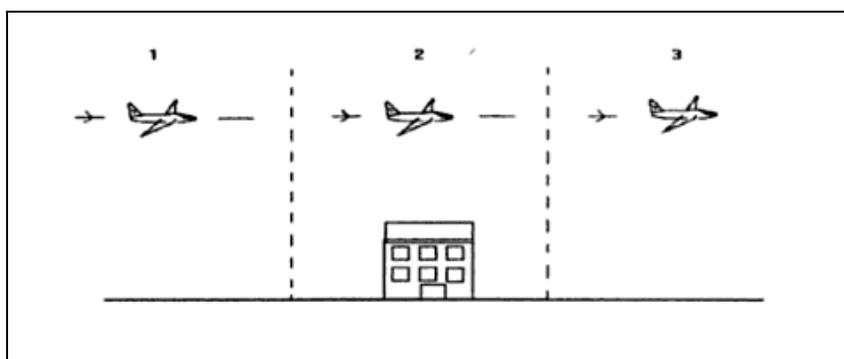


Figure 8. Example stimuli from Coventry and Mather (2002)

Different languages capture different distinctions between spatial scenes. For example, Dutch differentiates between attachment by a point versus a surface – both of which are instances of *on* in English (the clothes are *on* the line and the cup is *on* the table respectively). Spanish, on the other hand, has a single preposition *en* which covers all situations that would be covered by both English *in* and English *on*. Other languages rely more on verbs than prepositions to capture the meaning in a spatial scene. Figure 9 shows how different languages divide the containment-support continuum. This variation between languages is interesting from a cognitive psychology perspective since it raises questions about what aspects of spatial cognition are innate and which are culturally influenced. Since spatial language is grounded in perception, it makes sense that some part of the relationships we extract could be common between cultures and languages. At the same time there is clearly much variation.

Cross-linguistic variation is also interesting from a computational perspective since models of spatial preposition use should be flexible enough to learn prepositions from different languages.

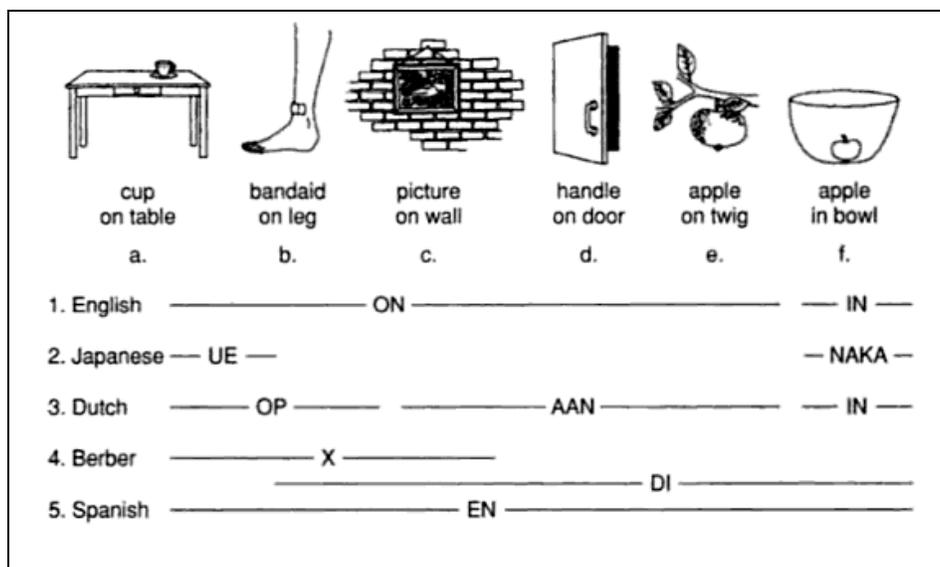


Figure 9. Support and containment prepositions in various languages. This continuum was used in a study by Bowerman and Pederson (in press) and was referenced in (Bowerman and Choi, 2001)

In an attempt to account for many of the findings described above, Coventry and Garrod (2004) have developed what they call the *functional geometric framework*. Their theory takes into account not only the findings on how people see the world, but also how people interact with objects and how objects interact with each other. The functional geometric framework is composed of three basic components: geometric routines, dynamic-kinematic routines and object knowledge (where the latter two together can be referred to as extra-geometric information). The geometric routines in the functional geometric framework encode “where” objects are located in the world. Dynamic-kinematic routines encode “how” the objects are interacting or how they may interact. The object knowledge captures how “what” objects are may influence our perceptions of “where” they are. Describing their theory, they point out that part of the role of spatial prepositions is not just to indicate where a given object is at a certain time, but also how likely it is to remain in that location in the future. Knowing how

objects interact and how likely they are to interact has an important influence on how we describe their spatial relationship.

This recent work from psychology, including the functional geometric framework, has strong implications for how computational models of spatial preposition use should work. Models that hope to capture the complexities of human preposition use cannot rely solely on simple geometric relationships. They need to incorporate knowledge about objects in the world and their typical roles. Models will also need to be able to use qualitative physics to understand the potential kinematics of given arrangements of objects. Since spatial language plays such an important role in human cognition and activity, having good artificial intelligence models of its use is critical to many tasks. Domains from GPS/GIS to image processing to game playing all critically need a human-like understanding of spatial language to be effective. Hence, spatial language modeling has also been an area of active research in the artificial intelligence community. The richness of spatial language makes computational modeling of it both algorithmically and computationally complex. Various AI groups have tried different approaches to this problem. Mukerjee (1998) provides a roadmap of the types and techniques of a number of spatial language models. Any model necessarily makes tradeoffs between breadth and depth of coverage – how many prepositions will be covered, how many factors will be considered, how varied will the stimuli be, etc. Chapter 4 of this thesis describes several models of spatial preposition use, and positions them within the context of these other models of spatial language while this section has provided the psychological background necessary to motivate and inform this work.

2.3 MULTIMODAL KNOWLEDGE CAPTURE

Multimodal knowledge capture (learning from multimodal sources) happens when people are presented with information in more than one modality. Traditionally, textbooks have been the primary source of multimodal learning, presenting students with combinations of diagrams and written

language, but the term also covers other multimodal sources of information like animations with written text or animations with spoken dialogs. The case for presenting material in multiple modalities is that people can understand an explanation better when presented with words and pictures than when presented with words alone (e.g. Mayer, 1989; Mayer and Gallini, 1990). A number of studies have explored these effects, the conditions under which they occur, and the role that individual differences play in their appearance (e.g. Mayer, 2001; Hegarty and Just, 1993; Larkin and Simon, 1987).

In a series of experiments, Richard Mayer and colleagues (summarized in Mayer, 2001), examined whether learners perform better on retention and transfer tests when they learn from multimodal sources of information than from single modality sources. Retention tests involve being able to recall information that was in the presented materials. Transfer tests involve being able to use that information to solve novel problems. In six of nine experiments subjects performed better on retention tests when they learned the original material from a multimodal source than when they learned from a single modality source. In all nine of those experiments, the multimodal subjects performed better on transfer questions. The retention results are even more impressive if only the results from book-like combinations of text and diagrams are considered – the multimodal subjects outperformed the single modality subjects on 5 out of 6 of those experiments (the other experiments involved animation + audio narration). There is also evidence that multimodal effects are stronger on delayed-recall tests than on immediate-recall (e.g. Peeck, 1989; summarized in Levie and Lentz, 1982).

In addition to studying if multimodal effects occur, learning scientists are interested in how and why people learn better from multimodal sources. Previous theories have suggested that multimodal sources are more useful for learning simply because the information is presented multiple times, reinforcing the concepts. Another potential explanation is that since the material is presented in multiple modalities, each learner can attend to the one that they best learn from. However, more

recent research focuses on the role that *active learning* plays in the multimodal effect. Active learning occurs when learner must engage cognitively with the material in an attempt to build a coherent understanding. Mayer's *multimedia learning theory* (2001) claims that multimodal sources of information encourage this kind of active engagement and that engagement may be what leads to the multimedia effect.

Mayer's theory relies on several underlying assumptions about human thought and learning. The first assumption is that of active learning. The second assumption is that people have *dual channels* for processing incoming information. There are separate channels for auditory and visual information, where auditory and visual are defined in terms of presentation mode (i.e. whether the stimulus is verbal (words) or non-verbal (pictures)). This is similar to Paivio's dual coding theory (Paivio, 1986). The third assumption is that human working memory has *limited capacity*, i.e., we are limited in the amount of information that we can process in each channel at one time. Limited capacity forces learners to make *metacognitive* decisions about what information to attend to and which connections between different pieces of information are worth building.

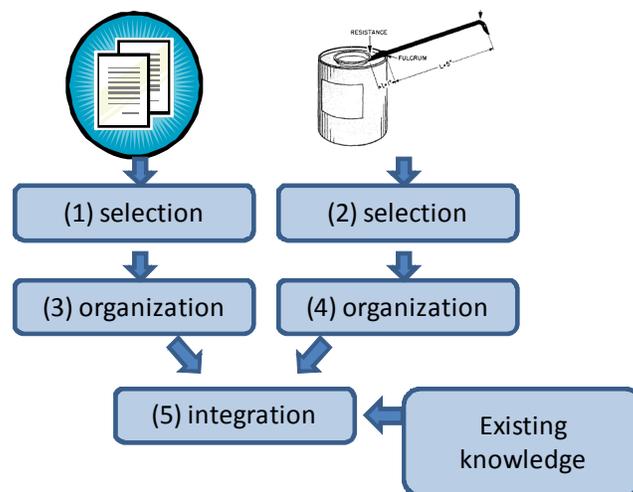


Figure 10. The five steps in Mayer's multimedia learning theory

There are three cognitive processes that make up Mayer's theory of multimedia learning: selecting relevant material, organizing selected material, and integrating selected material with existing knowledge. These three processes can be further broken down into five distinct steps as shown in Figure 10.

- (1) selecting relevant words for processing in verbal working memory
- (2) selecting relevant pictures for processing in visual working memory
- (3) organizing selected words into verbal mental model
- (4) organizing selected pictures into visual mental model
- (5) integrating verbal and visual representations along with prior knowledge

The key step is (5) integrating the verbal and visual representations along with prior knowledge. In this step the learner goes from two separate representations to one integrated representation in which "corresponding elements and relations from one model are mapped onto the other" (Mayer, 2001).

"When the handle is pulled up, the piston moves up, the inlet valve opens, the outlet valve closes, and air enters the lower part of the cylinder."

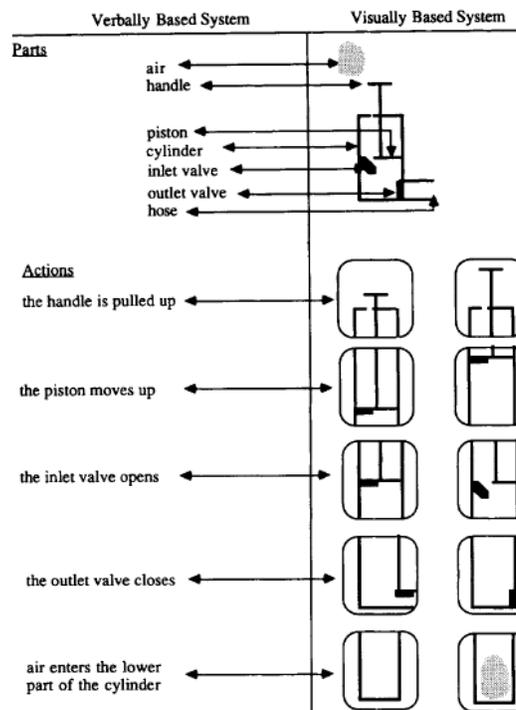
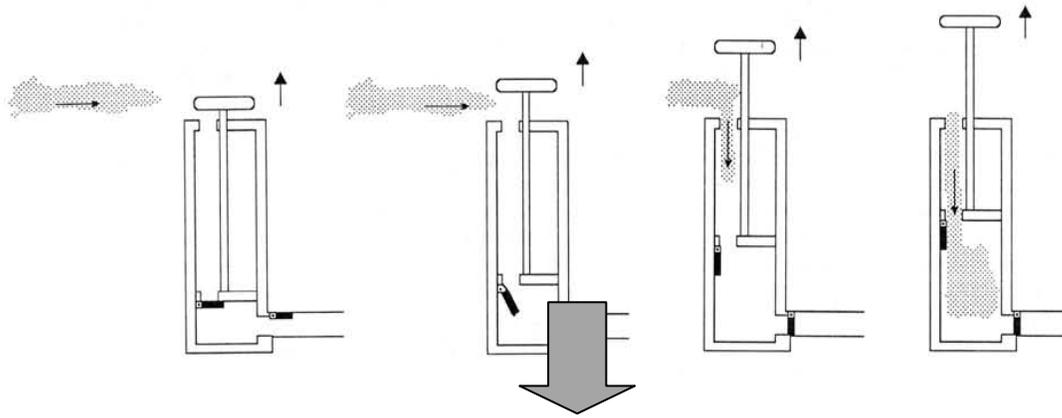


Figure 11. An example of multimedia learning in action, showing how the original material (top) is partitioned into selections by the learner (bottom) (compiled from two figures in Mayer & Simms, 1994)

Figure 11 is an example demonstrating Mayer's multimedia learning theory (2001). This example shows how a simple pump works. The figure illustrates how the learner breaks the text and diagram into

segments for integration (steps 1 and 2 of the multimedia learning process). Studies with human subjects lend support to a theory of incremental integration. For example Hegarty and Just (1989) found that subjects viewing text and diagrams of a pulley system looked back and forth between the text and the diagram multiple times while reading a selection.

There are multiple other variables that can affect how the presence of diagrams impacts learning. Learner ability, both in the subject matter (Mayer and Gallini, 1990; Mayer *et al*, 1995) and spatial ability more generally (Mayer and Simms, 1994), has been shown to influence how diagrams are used. The proximity of diagrams to their accompanying text (Mayer, 1989) and the inclusion or exclusion of distracting material (Mayer *et al*, 1996; Harp and Mayer, 1997) can also impact learning. Beyond the placement of diagrams/illustrations, the content also matters.

Consider the two diagrams from *Basic Machines* shown in Figure 12. The diagram/text pair on the left serves to illustrate the parts of a lever and the text refers directly to the labels in the diagram, highlighting important relationships. The text/diagram pair on the right serves a very different purpose, it is more illustrative, or entertaining, than informative, it illustrates a higher-level concept. One challenge of multimodal knowledge capture is to not only learn to extract information from diagrams, but to also be able to distinguish between different types of diagrams and how to tailor the type of information extracted to the type of diagram. For example, a system reading *Basic Machines* should invest much more time in extracting and understanding the spatial relationships in the diagram on the left in Figure 9 than the diagram on the right.

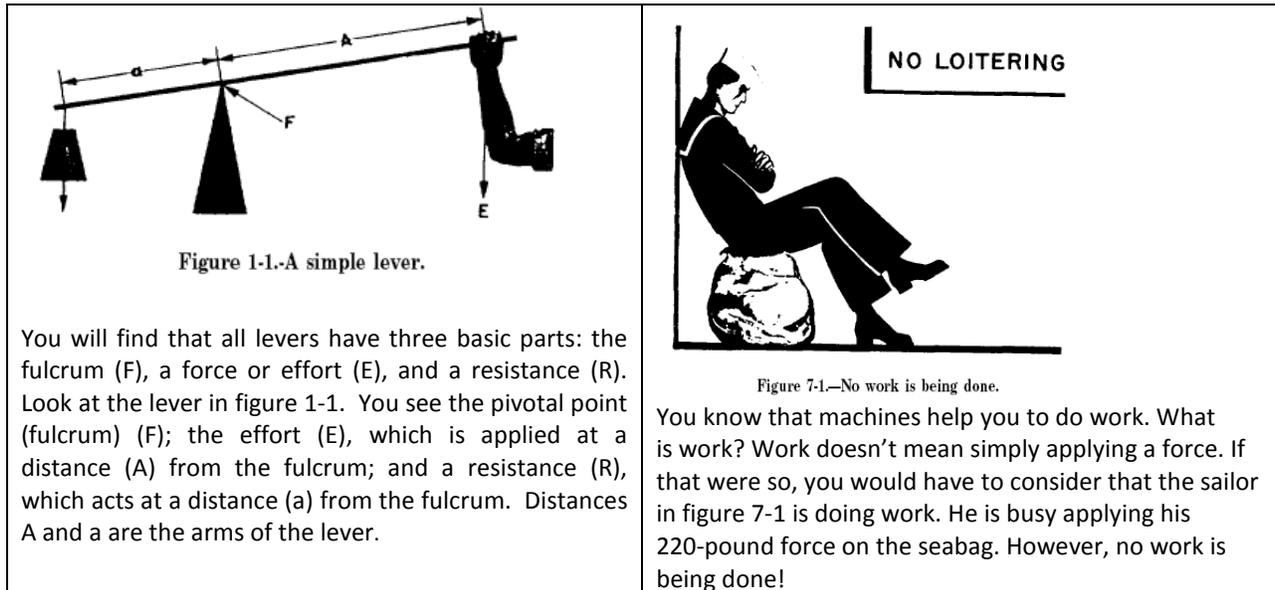


Figure 12. Two text/diagram pairs showing different uses of diagrams in text.

The multimedia theory of learning described here serves as a rough guide for the MMKCap model in Chapter 5. MMKCap owes many of its high-level design choices to this theory: text and diagrams are selected into individual chunks, the text chunks and diagram chunks are processed separately, and then they are integrated based on mappings between them. MMKCap, like Mayer's subjects should be able to learn and recall information from multimodal knowledge sources. However, it does not address some of the finer-grained distinctions such as specific working memory constraints, or coding modalities.

3 SYSTEMS BACKGROUND

3.1 INTRODUCTION

This chapter focuses on the existing and evolving systems that form the basis for several of the experiments in this thesis. In Artificial Intelligence and Cognitive Modeling, there is a temptation to build a system or model once to explain a particular phenomenon and then to move on to the next project. While this strategy allows for exploring a variety of different ideas, the systems developed are often short-lived and overly-tailored to one specific set of circumstances. An alternate approach, and the one in this work, is to work with a set of existing broad-domain models or systems that have been tested using a variety of input and to build on top of them. The benefit to this approach is that the component systems have been rigorously tested and their broad usage means that they cannot be tailored to facilitate the specificities of a given experiment. As additional experiments are done, new information can be fed back into the original systems, informing future development and strengthening the platforms for future work. The systems described in this chapter have been used in various configurations to simulate everything from perception to higher-level reasoning.

In this chapter, I first describe the large common sense knowledge base and the FIRE reasoning engine that reasons over it. Next, I discuss the Structure Mapping Engine (SME) model of analogy and similarity and the SEQL model of analogical generalization which uses SME to model category formation. Then I describe the EA NLU natural language understanding system and conclude by introducing the CogSketch open-domain sketch understanding system. Both EA NLU and CogSketch are used as input modalities for different experiments in this thesis. Relying on open-domain systems to create representations reduces the tailorability of experimental inputs and is much more time and effort efficient than creating representations by hand. This chapter gives general background on these

systems while chapters 4 and 5 describe their uses in spatial language and multimodal knowledge capture experiments.

3.2 LARGE COMMON SENSE KNOWLEDGE BASE AND THE FIRE REASONING ENGINE

3.2.1 COMMON SENSE KNOWLEDGE BASE

There is a large amount of commonsense information that humans rely on when they interact with the world around them. This information is often so basic that it is almost never explicitly stated in day-to-day human life (e.g. “all dogs are animals, but not all animals are dogs” or “a car pool is not a place to swim inside a vehicle”) but need to be articulated for artificial intelligence systems. AI systems need this kind of knowledge so that they can engage in the same types of reasoning tasks that people do on a daily basis. In AI, this information is provided in knowledge bases (KBs) which are collections of structured facts and axioms that codify knowledge about the world. The ResearchCyc Knowledge Base is a product of the Cyc project (Lenat, 1995) whose goal is to build a knowledge base that captures a broad selection of commonsense background knowledge. The work in this dissertation uses a subset of the ResearchCyc KB containing 2,182,300 facts and 14,257 relations, including a small number of extensions related to natural language understanding and QP theory (Forbus, 1984). Using externally developed ontologies is one way in which I have reduced the tailorability of my experiments.

The knowledge in ResearchCyc is expressed in CycL (Matuszek *et al.*, 2006), a language based on predicate calculus. In CycL, constants denote specific individuals or collections. For example, `Dog` could denote the collection of all dogs and `DogTraining` all events where a dog is trained. An individual might be a tangible individual such as `RoverTheDog`, or an individual could be a relation like `likesAsFriend` or `ownsPet`. CycL also uses formulas in which a relation is applied to some arguments. Sentences are well-formed formulas with a relation in the first position such as

(likesAsFriend DickCheney JonStewart). The other type of CycL formula is a non-atomic term. These have a logical function (i.e. a predicate which is an instance of Function-Denotational) as the first term, for example (FruitFn AppleTree) or (GovernmentFn France). This brief overview of CycL should be enough to understand the examples in this thesis. For a more thorough explanation of CycL conventions, please see the online documentation¹. ResearchCyc uses a *microtheory* structure to provide contextualization of facts – every fact must be stored in one or more microtheories. This allows for the consistent coexistence of facts which are contradictory at face-value. For example, a fact stating that dinosaurs rule the earth is true in the context of the Jurassic period, but is currently false.

3.2.2 THE FIRE REASONING ENGINE

FIRE is the reasoning engine used in this work to organize and access KB contents. FIRE is designed to support building general-purpose reasoning systems operating over large knowledge bases (in this case drawn from ResearchCyc). As in Cyc, microtheories are used to specify the logical environment for reasoning. FIRE is designed from the ground up to support analogical processing (via the Structure-Mapping Engine which is described in the next section). Analogy in FIRE is considered to be more primitive than even backchaining.

The working state associated with a system using FIRE is stored in one or more *reasoners*. Reasoners include a *working memory* that stores the assumptions and results specific to a particular session or use of an application. The working memory in FIRE is implemented using a version of the LTRE from Chapter 10 of (Forbus & deKleer, 1993). In this dissertation, FIRE is used to perform analogical mapping and to retrieve and store information in the knowledge base.

¹ http://www.cyc.com/doc/tut/ppoint/CycLsyntax_files

3.3 ANALOGY AND SIMILARITY: SME, SEQL AND MAC/FAC

3.3.1 SME

The Structure-mapping Engine (SME) (Falkenhainer, Forbus & Gentner, 1989) is a model of analogy and similarity based on Gentner's (1983) Structure-mapping Theory. In structure-mapping, analogy and similarity are defined in terms of a structural alignment process operating over structured, relational representations. SME takes as input two cases, a *base* and a *target*. Each input case is a structured representation consisting of entities, attributes, and relations. Given the two cases, SME produces one to three mappings between them by aligning their common structure, with the goal of constructing the maximal structurally consistent match. A structurally consistent match is one that satisfies the following three constraints: *tiered-identity*, *parallel connectivity*, and *one-to-one mapping*. Tiered-identity enforces a strong preference for matches only between identical predicates, but allows for rare exceptions (e.g. matching aligned functions in cross-domain analogies). For example, *minimal ascension* (Falkenhainer, 1988) allows non-identical predicates to match, but only if they are part of a larger mapped structure and share an ancestor in the ontological hierarchy. Parallel connectivity states that if two statements are matched then their arguments must also match. One-to-one mapping means that each element in the base can align with no more than one element in the target and vice versa.

Each mapping in SME contains: (1) a set of correspondences between elements in the base and elements in the target, (2) the *structural evaluation score* which is a numerical measure of similarity and (3) *candidate inferences* which are inferences that are carried over from the base to the target, according to common structure. Structural evaluation prefers mappings which align higher-order relations; this is the principle of *systematicity*. SME has been used to model many phenomenon, including perceptual similarity (Lovett, Deghani & Forbus, 2008), learning of physics concepts from

examples (Klenk & Forbus, 2007) and moral decision making (Dehghani *et al.*, 2008). In this work, SME is used in question answering, which serves as the evaluation for the techniques of multimodal knowledge capture.

3.3.2 SEQL

In the spatial language experiments in Chapter 4, I model category formation using analogical generalization via SEQL (Skorstad, Gentner, & Medin, 1998; Kuehne *et al.*, 2000; Halsted & Forbus, 2005) which uses SME as a component. SEQL creates *generalizations* from an incoming stream of examples. In supervised learning experiments, generalizations are organized in *generalization contexts*. For example, in learning language categories, there would be one generalization context per word/label. There can be more than one generalization per context, since real-world concepts are often messy and hence disjunctive. Each generalization context consists of a set of generalizations and a set of unassimilated *exemplars*.

When a new example arrives, it is compared against every generalization in turn, using SME. If it is sufficiently close to one of them (as determined by the assimilation threshold), it is assimilated into that generalization. The probabilities associated with statements that match the example are updated, and the statements of the example that do not match the generalization are incorporated, but with a probability of $1/n$, where n is the number of examples in that generalization. If the example is not sufficiently close to any generalization, it is then compared against the list of unassimilated exemplars in that context. If the similarity is over the assimilation threshold, the two examples are used to construct a new generalization, by the same process. An example that is determined not to be sufficiently similar to either an existing generalization or unassimilated example is maintained as a separate example.

SEQL has been used to model everything from infant learning (Kuehne, Gentner & Forbus, 2000) to conceptual change (Friedman & Forbus, 2008) to generating rules for proposing perpetrators of

terrorist activities (Halstead & Forbus, 2007). In this work it is used to model the formation of spatial language categories.

3.3.3 MAC/FAC

MAC/FAC (Many are Called/Few are Chosen) (Forbus, Gentner, & Law 1995) is a model of similarity-based reminding created to capture the seemingly contradictory psychological phenomenon that surface similarity is often more important than structural similarity in retrieval, while structural similarity is weighted more heavily in similarity judgments. MAC/FAC takes as input a *probe* and a *case library*. The probe contains the structured representation for the situation under consideration (for example, a physics problem to be solved). The case library is a set of cases, each itself a structured representation, representing the available examples to match (for example, a set of previously worked physics problems). MAC/FAC uses a two-stage process to select a *reminding* case from the case library based on its similarity with the probe.

In the first stage (MAC) a feature vector is created for the probe. The components of the vector correspond to individual predicates and have a value that is proportional to the number of occurrences of that predicate in the case. The case(s) from the case library with the highest dot product with the probe is returned from MAC. Up to three cases can be returned, based on how close the dot product results are (this threshold is a parameter that can be set by the user). The second stage (FAC) uses SME to do a more detailed, structural comparison of the probe with each of the candidate cases returned from the MAC stage. The candidate case with the highest structural evaluation score is returned as the reminding by MAC/FAC. Alternatively, if the scores are extremely close, MAC/FAC may return up to three reminders.

3.4 EA NLU

The Explanation Agent Natural Language Understanding (EA NLU) (Tomai & Forbus, 2009) is used to create representations from English language input for the experiments in this thesis. Unrestricted automatic natural language understanding is beyond the state of the art, so EA NLU uses a simplified language, QRG Controlled English (QRG-CE), and relies on semi-automated disambiguation. This practical approach is a significant improvement over having experimenters construct representations entirely by hand, which is time consuming and allows for increased tailoring of inputs. The EA NLU approach also allows us to make the syntax problem tractable, to build deep representations suitable for complex reasoning, and to handle a wide range of potentially ambiguous inputs. EA NLU is used by multiple projects. Experiment-specific alterations and additions are explained on an experiment by experiment basis.

EA NLU relies on several off-the-shelf components. It uses Allen's bottom-up chart parser (Allen, 1995) in combination with the COMLEX lexicon (Macleod *et al.*, 1998) and a simplified English Grammar (Kuehne & Forbus, 2004). The parser uses subcategorization frames from ResearchCyc for word and common phrase semantics. Compositional frame-based semantics from the parsing process are transformed using dynamic logic principles from Discourse Representation Theory (DRT) (Kamp & Reyle, 1993). The resulting set of *discourse representation structures* (DRS) supports numerical and qualitative quantification, negation, implication, modal embedding, and explicit and implicit utterance sub-sentences. EA NLU has been used previously to create representations for models of moral decision making (Dehghani *et al.*, 2008). In this work, EA NLU is used to create representations of input for a multimodal knowledge capture system.

3.5 COGSKETCH

CogSketch² (Forbus, *et al*, 2008) is an open-domain sketch understanding system, built on the nuSketch (Forbus, Ferguson, & Usher, 2001) architecture. CogSketch is being developed for three main applications: (1) as a cognitive simulation of visual and spatial reasoning and learning, (2) as a platform for collecting data from human subjects, and (3) as an educational tool. In this thesis we use it in the first capacity, as a means of creating input stimuli for cognitive modeling and artificial intelligence research. A previous version of CogSketch, called sKEA, was used for some of the experiments reported. For the mode of sketching used, the fundamental properties of the sketches and the spatial relationships computed are the same in the two programs; the main difference is the underlying knowledge base: CogSketch uses a subset of OpenCyc³ while the sKEA KB is based on ResearchCyc³.

The main insight in CogSketch is that recognition is not a necessary aspect of human to human sketching. People often use language to label their sketches in real time instead of relying on themselves or others to recognize the objects drawn (think of the ubiquitous cocktail napkin sketch). This is a key insight, as recognition-based sketching systems (e.g. (Alvarado, Oltmans, & Davis, 2002)) must restrict themselves to tightly-controlled domains with a relatively small number of highly-differentiated symbols in order to operate correctly. CogSketch operates as a general purpose sketching system and bypasses the problems associated with object recognition by providing users with tools to segment and label their own ink. Users segment sketches into distinct objects as they sketch by clicking a button when they start drawing an object, and again when they finish. Each object created is a *glyph*. Each glyph has *ink* and *content*. Ink consists of one or more polylines, lists of points representing what the user drew. The content is a symbolic token that represents what the glyph denotes. *Conceptual*

² <http://www.spatiallearning.org/>

³ This difference is due to licensing. CogSketch is available for free download, so it uses the unrestricted OpenCyc knowledge base.

labeling allows users to indicate the type of the content of the glyph in terms of the underlying knowledge base. In addition to a conceptual label, glyphs in CogSketch can be given a *name* which is a natural language string used to identify the glyph. Figure 13 below shows an example of a CogSketch sketch. There are three glyphs in this sketch, the sun, the planet and the orbit. The glyph representing the planet is named “Earth” and conceptually labeled with the ResearchCyc concept `Planet`, therefore in later reasoning it can be identified by its name “Earth”, and the system will be able to access everything that ResearchCyc knows about the concept planet.

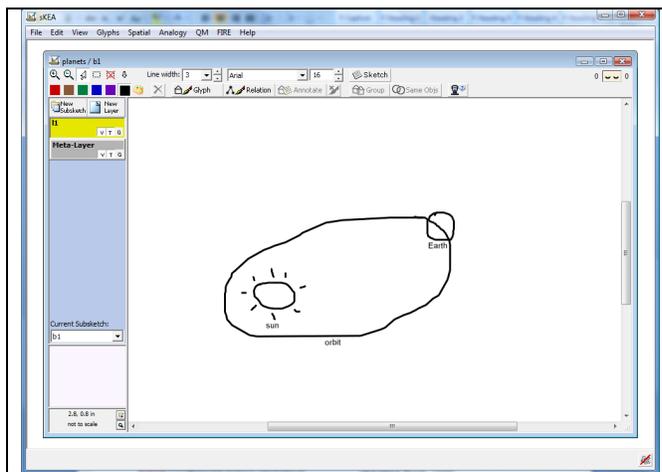


Figure 13

A CogSketch sketch containing three glyphs: a planet, a sun, and an orbit. The glyphs representing the planet and the orbit form a connected glyph group while the glyphs representing the orbit and the sun form a contained glyph group with the orbit filling the role of the container.

CogSketch also automatically computes a number of qualitative visual relations and attributes for glyphs in sketches. These represent the general visual features of the sketch and therefore do not make use of any task or domain specific information. For example, a glyph’s size is computed based on the relative size of its bounding box. The bounding box and blob boundary are automatically computed for each glyph. The RCC-8 qualitative relations (Cohn, 1996), which describe all possible topological relations between two-dimensional shapes, are computed between all pairs of glyphs. RCC-8 relations are used to guide the generation of other relations, including positional relationships and containment and connection.

The visual positional relationships that are computed depend on the *genre* and *pose* of the sketch under consideration. Genre is used to tell CogSketch the overall type of what is being drawn, while pose tells CogSketch how to interpret the sketch coordinates in terms of the kind of thing being drawn. The sketches in this dissertation are all drawn in the default genre, *physical*, and the default pose, *looking from side*. In this genre/pose, the positional relations computed are: *rightOf*, *above*, *enclosesHorizontally*, and *enclosesVertically*. Containment and connection trigger the creation of *contained glyph groups* and *connected glyph groups*. Contained glyph groups consist of a single container glyph and all of the glyphs contained inside of it. Connected glyph groups consist of all glyphs whose ink strokes intersect. In the sketch in Figure 13, the planet and the orbit form a connected glyph group since their ink overlaps. The orbit and the sun form a contained glyph group with the orbit as the container.

All of the relations we have discussed up to this point are done on or between glyphs. It is also possible to decompose glyphs in CogSketch into their component edges using the *Perceptual Sketchpad* (PSketch)(see Lovett, Deghani, & Forbus, 2008 for a more detailed description). Decomposition is done over the polylines that make up a glyph. Starting with each polyline as a candidate edge, PSketch uses a five step algorithm to refine its list of candidate edges. Once PSketch has identified the edges in a glyph, it attempts to find cycles of edges that potentially form a closed shape. PSketch then computes a qualitative structural representation based on the relationships between the edges

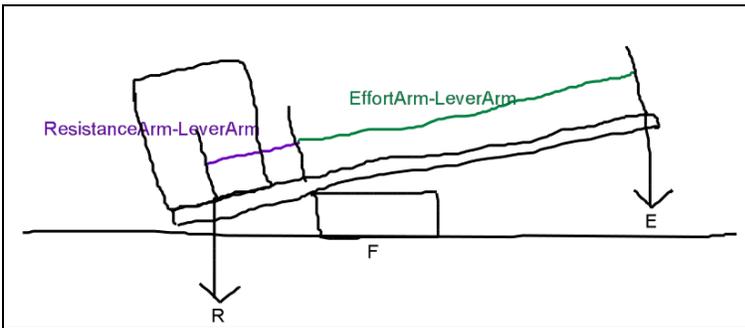


Figure 14
Sketch showing the use of annotation glyphs. The Resistance Arm is an annotation glyph of type LengthIndicator and indicates an important property of the lever glyph that it annotates.

In addition to standard glyphs, CogSketch allows users to create *annotation glyphs*. Annotation glyphs provide a means of highlighting important properties of a glyph. Examples include the length or height of an object, or the force acting on an object. Like other glyphs, annotation glyphs consist of both ink and the entity that is represented by the glyph (in this case chosen from a list of properties of glyphs). Unlike other glyphs, annotation glyphs also refer to the other glyph in the sketch depicting the entity that they are providing information about. Annotation glyphs may also have values and units associated with them. For example, in Figure 14, the Resistance Arm is an annotation glyph of type *LengthIndicator*, indicating the length of the Resistance Arm of the depicted lever. The Resistance Arm glyph has a slot referring to the lever glyph, since the lever is the object being described. The Resistance Arm glyph also has an associated value of 1 foot. Due to their special status, annotation glyphs do not participate in the computation of spatial relationships.

Since CogSketch is an open-domain system, it has been used in a variety of settings. CogSketch is being integrated with QR systems for use in the early stages of engineering design (Wetzel & Forbus, 2008). CogSketch has also been used to create input for a variety of cognitive modeling experiments (e.g. (Lovett, Lockwood & Forbus, 2008)). In this thesis CogSketch is used to create inputs for modeling spatial language use and to create diagrams for multimodal knowledge capture.

3.6 DISCUSSION

This chapter has described the existing component systems that I used in this thesis. Unless otherwise noted, all systems were developed in the Qualitative Reasoning Group at Northwestern University. First, I covered the ResearchCyc knowledge base and the FIRE reasoning engine which form the common-sense knowledge base for the other systems and my work. I also described the SME model of analogy and similarity and the related SEQL model of analogical generalization, both of which are used in my simulation experiments. Finally, I introduced the EA NLU natural language understanding system and the CogSketch sketch understanding system, both of which are used throughout my experiments to create structured representations from natural language and sketched input. The descriptions in this chapter are necessarily brief and intended only to give enough background on each system to enable the reader to understand the experiments in this document. For theoretical and implementation details, the referenced papers will give more in-depth information. Any experiment-specific alterations or additions to existing systems are described in the section on that experiment.

4 SPATIAL PREPOSITION EXPERIMENTS

4.1 INTRODUCTION

Spatial relationships play an important role in human reasoning, from navigation to solving physics and engineering problems. Space is a fundamental organizing principle of human cognition and primitives to communicate spatial relationships show up across languages and cultures. In English-speaking countries we use the spatial prepositions (e.g. *in*, *on*, *above*, *below*) to talk about the spatial arrangement of objects in our environment. While the number of prepositions in English is a closed set and is quite small (around 100) compared to sets of other word types, the assignment of these prepositions to actual scenes is quite complex. Psychologists have made significant progress in determining the scene characteristics that contribute to spatial preposition assignment, yet there are still many open questions. For a more in-depth coverage of the psychological factors influencing spatial preposition use, please see the discussion in Chapter Two. Computational models of spatial preposition use lag behind the psychological discoveries, and very few of these models have been applied to more complex reasoning tasks. In this chapter I present models of both the learning of spatial prepositions from sketches and the use of spatial prepositions to label relationships in novel sketched scenes.

For the first line of experiments, I present the SpaceCase model of spatial preposition use. SpaceCase is a Bayesian model of spatial preposition assignment that uses evidence from a scene to update its preference between *in* and *on*. I describe the architecture of the model and its success in modeling results from several psychological studies (Lockwood *et al.*, 2005). SpaceCase Experiment 1 examines how four properties of the figure and ground influence spatial preposition use based on the results of Feist and Gentner (2003). SpaceCase Experiment 2 models the role of spatial language in memory for visual scenes (Feist & Gentner, 2001).

In the remaining two lines of experiments, I model the formation of spatial preposition categories using analogical generalization. While others have modeled the formation of spatial categories, my approach is unique in that my system demonstrates the ability to learn spatial categories using a considerably smaller and more cognitively plausible number of training examples than previous models. I present results from categorizing sketches containing simple geometric figures (squares, circles, triangles). The two experiments in the first line in this section: Geometric Shapes Experiment 1 and Geometric Shapes Experiment 2 examine the categories created from generalizing canonical examples (in Experiment 1) and what happens to those categories when ambiguous cases are added (in Experiment 2).

In the final line of experiments, the stimuli involve real-world objects where functional aspects of the items must be considered in addition to scene geometry. With these stimuli, I also demonstrate the ability to use the same sketches and the same general processes to learn spatial prepositions from a second language. While some type of spatial language shows up across languages and cultures, the form that language takes differs greatly in both its specificity and the importance of different properties in the environment (e.g. Bowerman, 1996; Gentner & Bowerman, 2009). The Cross-linguistic Experiment classifies the stimuli from Gentner and Bowerman (2009) according to the English preposition labels and also classifies the same sketches according to the Dutch preposition labels. I conclude this chapter with a discussion of the results from all three lines of experiments and what they contribute to our understanding of computational models of spatial preposition use.

4.2 PROBLEM DESCRIPTION

There are two problems to tackle when modeling human spatial preposition use 1) category formation and 2) labeling of novel scenes. Category formation examines how spatial language categories are learned, including what properties of a scene are most important for distinguishing

category membership. Labeling of novel scenes is the problem of assigning a preposition to a scene (or a subset of a scene) which has not been encountered before, based on the categories formed. The SpaceCase experiments deal with the second problem – given existing category constraints, taken from the literature, can novel scenes be labeled in a manner consistent with human subjects? The other two lines of experiments address the first problem – can the category contents needed to classify future scenes be automatically learned? The learning experiments here do not currently produce output in a format that can be directly used by SpaceCase, however, ideas for implementing this in future work are discussed in Chapter 6.

4.3 SPACECASE MODEL OF SPATIAL PREPOSITION USE

This section describes the two experiments based on the SpaceCase model of spatial preposition use. SpaceCase was developed to model the phenomenon of spatial language use in a way that can account for psychological findings that show that functional features of objects are important to consider, in addition to the geometry of a given scene. We assume that the assignment of spatial prepositions rests on the knowledge and skills that people bring to bear on other spatial tasks. Spatial prepositions encode a combination of geometric and functional properties, making them both detectable in visual scenes and able to provide information about what possibilities are relevant in that scene when detected. For example, the distinction between *in* and *on* in English includes an aspect of location control, with more control when *in* is used than when *on* is used. Consider an apple *in* a bowl and an apple *on* a plate. When the bowl is moved the apple will always move with the bowl although it may roll around a little inside. The apple on the plate is in a much more precarious situation and may easily take a tumble if the plate is moved quickly.

We assume that there are multiple, situation-specific criteria that determine when it is appropriate to use one term over another – location control is just one example of such criteria. We

view each of these criteria as evidential, in that they tend to suggest rather than uniquely determine answers. Thus, we describe our SpaceCase model in terms of evidence rules, which given a situation, provide levels of belief about how prepositions should be assigned.

4.3.1 SPACECASE EXPERIMENT 1: LABELING

4.3.1.1 Overview

In SpaceCase Experiment 1, we focus on modeling the results of Feist and Gentner (2003). They examined the role of four factors in *in/on* determinations in visual scenes involving two objects: a *figure* object (located object) and a *ground* object (reference object). The four factors considered were: (1) the geometry (curvature) of the ground, (2) the animacy of the ground, (3) the animacy of the figure, and (4) the functional role of the ground. The curvature of the ground was varied to be one of three qualitative values {high-curvature, medium-curvature, low-curvature}. The animacy of the ground was varied by using either a hand (an animate ground) or a variety of inanimate objects as the ground figure. The animacy of the figure was varied by using either a coin (inanimate) or firefly (animate) object as the figure. The functional role of the ground was studied by varying the inanimate labels assigned to the ground from the set: {bowl, dish, plate, slab, rock}.

In the original study, subjects were shown simple pictures of figure objects located with respect to ground objects. The different factors (geometry, animacy, and function) were systematically varied in the different stimuli. Subjects were given sentences of the following format: “<figure> is *in/on* the <ground>” where <figure> and <ground> were replaced with the labels that matched the pictorial stimuli. The subjects were asked to indicate which preposition best fit the situation displayed in the stimulus.

All four factors studied were shown to affect preposition assignment. Specifically, high curvature of the ground was more likely to lead to *in* and low curvature is more likely to be associated with *on*. If the ground is animate (the hand), *in* was more likely to be used (presumably because the hand can close around the figure object and better contain it) whereas if the figure is animate (the firefly), *on* was used more often (because the firefly is able to move away from the ground of its own volition). Moreover, subjects were more likely to use *in* than *on* when they were told that the ground was a container (such as a bowl) than when they were told it was something that is usually considered to be a surface (plate or slab), even when the object exhibited the same level of curvature. While all of these factors influenced the use of prepositions, the strength of the effect varied.

4.3.1.2 Materials

The original stimuli from the Feist and Gentner (2003) study were used as input to the SpaceCase model. Each of the original stimulus objects was sketched using the CogSketch sketch understanding system. A total of six sketches were created: one of each variation on ground curvature (low, medium, high) with a firefly as the figure and one of each variation on ground curvature with the coin as the figure. All other variations were created by relabeling the ground object in one of the existing sketches. This made for a total of 36 stimuli accounting for each variation along the four dimensions (summarized in Table 2 below). Figure 15 shows an example of a stimulus from the original experiment along with the sketched equivalents.

Table 2. Variations of scene factors taken from Feist and Gentner (2003). An asterisk indicates an approximation taken from the ResearchCyc contents and a plus indicates a new concept that was created for this experiment.

Dimension	Variations	Determined by	ResearchCyc label
Geometry of the ground	high-curvature medium-curvature low-curvature	Ratio of height to width of the bounding box of the ground	N/A
Animacy of the ground Function of the ground	hand dish bowl plate slab rock	genls inferencing	Hand EatingVessel* Bowl-Generic DinnerPlate Block* StoneObject-Natural
Animacy of the figure	coin firefly	genls inferencing	Coin-Currency Firefly+

The sketched stimuli differed from the original stimuli in that they were 2-D renditions of the original 3-D stimuli. From CogSketch we were able to extract a variety of qualitative geometric properties directly from the ink and to use the conceptual labels to describe the necessary functional information (figure and ground animacy and function of the ground). SpaceCase collects information on each of the four factors under consideration from each input sketch. The geometry of the ground is computed from the properties of the ink in the sketch and the other three factors are collected via inference about the kinds of entities involved. These inferences are based on the conceptual labels assigned to the figure and ground objects in the sketches. For many of the stimuli, the knowledge base already contained the appropriate concept: `Hand` (hand), `Coin-Currency` (coin), `DinnerPlate` (plate), `StoneObject-Natural` (rock), and `Bowl-Generic` (bowl) were all taken directly from the existing KB contents. There wasn't a concept directly corresponding to dish, so that was approximated with the concept `EatingVessel`. Slab was approximated with `Block` which has many of the same

properties as a slab. The knowledge base did not have a concept for firefly, so a new concept `Firefly` was created based upon the entry for `Dragonfly`. Table 2 summarizes these labels. In all of the sketches, in addition to the conceptual labels given, the figure object was named “figure” and the ground object was named “ground”.



Figure 15. Example stimuli from the original study (left) and the sketched equivalent (right)

Table 3. Rules for determining the label for ground function

Label	Criteria
Strongly functions as a container	both <code>ContainerArtifact</code> and <code>Basin</code> in gens hierarchy
Weakly functions as container	<code>ContainerArtifact</code> in gens hierarchy AND <code>Smooth</code> NOT in gens hierarchy OR <code>HolderGripper</code> in gens hierarchy
Functions as a surface	Anything that does not fall into one of the other categories

Curvature information is approximated from the ink by examining the height to width ratio of the bounding box of the ground glyph in the sketch. This is a very rough approximation of curvature and works only in 2D situations such as this where the objects are viewed from a side perspective, and we know that they will be curves. Assignment to one of the three qualitative categories (high, medium, low) is based on numerical cutoffs for each category. Animacy of the ground and animacy of the figure are binary values and are based on gens hierarchy in the knowledge base. An object is considered to be

animate if it has the concept `Animal` somewhere in its genls hierarchy. The function of the ground is qualitatively assigned to one of three categories: strongly functions as a container, weakly functions as a container, or functions as a surface. These categories were created based on the human subjects data. Assignment to a category depends on the genls hierarchy. Table 3 shows the genls inferencing rules used to determine the function label for the ground object. These criteria were set to try to capture the phenomenon from the human data as closely as possible. For example, one hypothesis was that an animate ground (hand) produced more occurrences of *in* because it could close around the object and contain it, so `HolderGripper` was chosen as the genls target. The comment for the concept describes it as “[an instance of `HolderGripper`] can apply pressure to another object and thereby grip it in such a way that its motion is restricted”, which captures the spirit of the original data.

4.3.1.3 System Design

All of the information gathered about each input scene is fed as evidence into a Bayesian updating algorithm that assigns a probability that either of the prepositions (*in* or *on*) accurately describes the scene. We use Everett’s (1999) evidential rule engine, which in turn uses Pearl’s (1986) hierarchical updating algorithm. Evidence contributes to the support for a preposition based on the likelihood of that piece of evidence for that preposition. Again, these values are taken from the human subjects data. For example, Feist and Gentner showed that an animate ground led subjects to choose *in* over *on*. Consequently, in our model an animate ground is considered evidence that increases the likelihood that *on* is the correct preposition to describe the scene. Likelihood in this case is defined as:

$$\lambda_n = \frac{P(e | H)}{P(e | \neg H)}$$

After all of the evidence is considered and propagates through the model, if the likelihood of any preposition exceeds a threshold, that preposition is proposed as the correct descriptor for the scene. At present, the only options available are: *in-ContGeneric*, *on-Physical*, and *other-preposition*. The first two are formal predicates which are used in the knowledge base for covering a very large set of specialized cases, defined by a hierarchy of specialized predicates. For example, *in-ContGeneric* has thirteen specializations including different levels of location control (e.g. open versus closed containers) and *on-Physical* has two specializations corresponding to floating on a liquid and particles strewn over a surface.

Currently SpaceCase has a total of ten evidential rules. Three of the rules are used to describe the support relationship between the ground and the figure. This is an example of using CogSketch to provide perceptual information to the system, since the triggers for these rules depend on the visual relationship between the sketched figure and ground relationships. The three support relationship rules are:

- *figure-completely-supported-by-ground*
- *figure-partially-supported-by-ground*
- *figure-not-supported-by-ground*

These rules trigger based on how many of the figure's bottom edge points intersect with the ground's edge points. The first rule (complete support) increases the likelihood that the figure is either *in* or *on* the ground. The second and third rules (no support or only partial support) increase the probability that another preposition would be more likely to describe the scene. For SpaceCase Experiment 1 (labeling), all of the figure ground pairs were in complete support relationships in keeping with the original stimuli.

The other seven rules in the SpaceCase system collect the evidence necessary to make *in/on* judgments based on the results from the Feist and Gentner (2003) study. Therefore, they are related to

the four factors that were studied in those experiments (animacy of the figure and ground, ground function and ground geometry). The variables that represent these likelihoods are:

- ground-high-curvature
- ground-medium-curvature
- figure-animate
- ground-animate
- ground-function-container-strong
- ground-function-container-weak
- ground-function-surface

Based on the results from the human subjects trials, the ground high and medium curvature rules increase the likelihood that the figure is *in* the ground. The ground animate and ground function container rules also increase the likelihood of *in*. The figure animate and ground function surface rules increase the likelihood that the figure is *on* the ground. The curvature rules are triggered by the curvature that CogSketch computes from the digital ink for the glyph that represents the ground. The other rules are all triggered by inferences made from the knowledge base, using the concept instance information asserted when the glyphs are drawn as discussed in the Materials section.

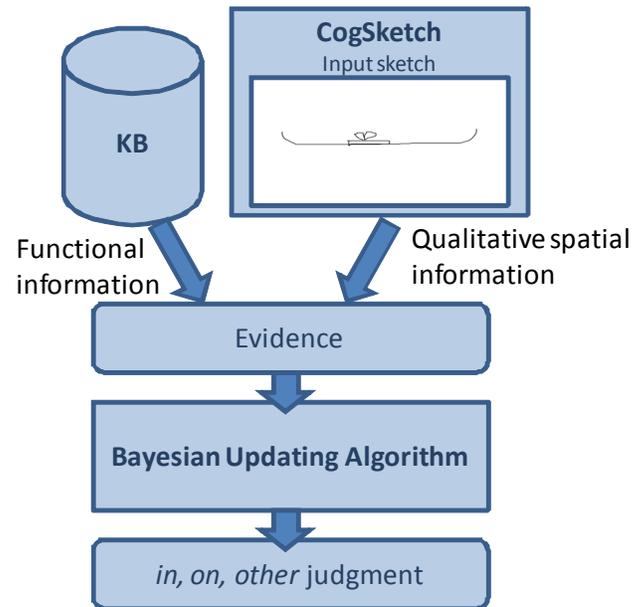


Figure 16. SpaceCase model

Figure 16 shows how information propagates through the SpaceCase model. When each rule is triggered, it creates an evidence element that contains the name of the preposition to update (*in*, *on* or *other*) as well as the amount by which to update its likelihood. The evidence values associated with each rule are parameters of the model. The values chosen were based on the pattern of results found in the human subjects experiments. For example, the function of the ground, for people, has a much stronger influence on the number of *in* responses than the curvature of the ground. Therefore, the ground function strong rule increases the likelihood of an *in* response by a greater value than the ground high curvature rule. The values of the likelihood update variables in the current incarnation of SpaceCase are described in Table 4 below. Later we will see that SpaceCase is not terribly sensitive to the specific values of these parameters. As long as their ordinal relationships fit the pattern of results found in the original study, SpaceCase's answers will accurately model the data.

Table 4. SpaceCase rules along with the preposition they support and their likelihood values

Variable Name	Preposition	Likelihood
figure-complete-support	in/on	5
figure-partial-support	other	10
figure-no-support	other	30
ground-high-curvature	in	3
ground-medium-curvature	in	2
figure-animate	on	3
ground-animate	in	5
ground-function-container-strong	in	10
ground-function-container-weak	in	5
ground-function-slab	on	10

4.3.1.4 Results

SpaceCase was consistent with human subjects on all 36 experimental stimuli for the values of the parameters given in Table 4. Importantly, SpaceCase is not overly sensitive to the specific values chosen for the likelihood parameters. As long as the parameters reflect the relative strengths of the factors as found by Feist and Gentner (2003), the correct results are derived. We determined this via a series of sensitivity analyses, looking at how the results changed when parameters were varied. Because this is a large space, we have focused on two-dimensional subspaces of these parameters at a time, with the other parameters keeping the values from Table 4. Figure 17 is an example plot showing the 2x2 sensitivity analysis for the figure-animacy and ground-container-strong parameters. The lighter gray squares indicate parameters settings where SpaceCase's answers are consistent with human subjects and the darker square indicate inappropriate results.

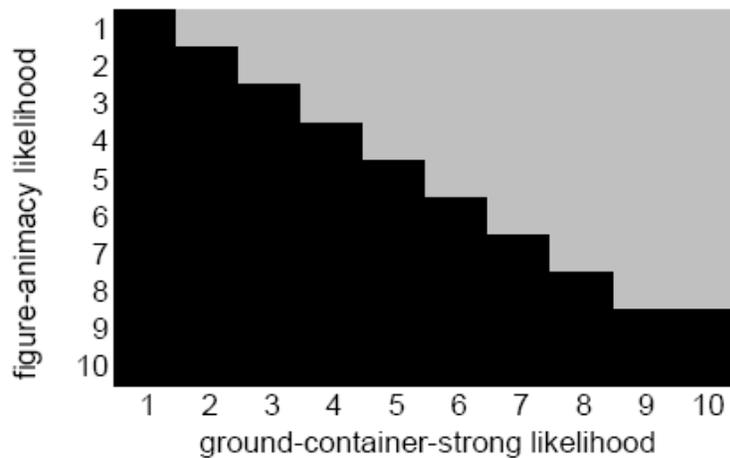


Figure 17. Example of 2x2 sensitivity analysis run on the SpaceCase likelihood values

Examining the instances that give inappropriate answers can lead to interesting insights. For example, the firefly-hand stimulus proves to be particularly interesting since both are animate and an animate ground should bias the response towards *in* while an animate figure should bias responses towards *on*. Subjects from the original study were in fact more likely to respond with “the firefly is *in* the hand,” and Feist and Gentner (2003) reported a much larger positive effect on *in* usage for ground animacy than the effect for *on* generated by figure animacy. SpaceCase shows the same pattern of results, providing *in* as the answer unless the figure-animate parameter is set sufficiently higher than the ground-animate parameter in which case it returns *on*. Thus, when SpaceCase’s parameters violate constraints found by psychological experimentation, it fails, suggesting that it is failing for the right reasons.

4.3.2 SPACECASE EXPERIMENT 2: THE EFFECT OF SPATIAL LANGUAGE ON RETRIEVAL

4.3.2.1 Overview

SpaceCase was also used to model a second set of spatial preposition experiments showing the effects of spatial language on memory. Feist and Gentner (2001) describe a series of experiments where human subjects were shown pictures that were ambiguous as to whether or not the figure was in a particular spatial relationship with the ground. In some trials, subjects were also shown a sentence describing the scene using spatial prepositions. For example, in Figure B (initial picture) it is ambiguous as to whether the block is *on* the building. The experimental group in the original study might be asked to rate the applicability of the sentence “the block is on the building” while being shown the picture (there were several variations of the experiment to rule out alternate explanations such as concentrating on the picture, and language without prepositions), and the control group would see the picture without any language. Later, subjects would be shown pictures and were asked to pick out the stimuli that they had originally seen. In the retrieval phase subjects would see the original picture, as well as two variations, a plus variant which unambiguously satisfies the spatial preposition and a minus variant which is even worse in regards to exhibiting the spatial preposition than the original stimulus. All three of these variations are shown in Figure 18 below. Subjects tended to believe that they had seen the plus variant when they were also exposed to the appropriate spatial language during encoding, thus illustrating that language can affect the encoding and memory of spatial relations in visual stimuli.

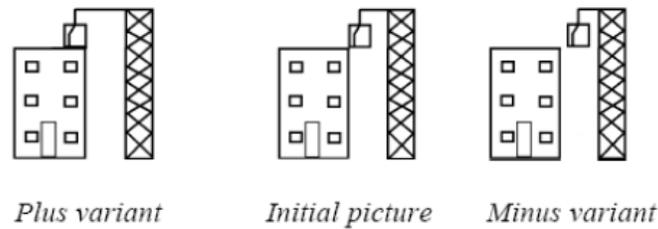


Figure 18. Example Stimulus from the original Feist and Gentner study

4.3.2.2 Materials

To model these results, we recreated all of the stimuli (original pictures as well as plus and minus variants) using CogSketch. Because SpaceCase currently only handles *in/on* distinctions where the figure is supported by the ground, we eliminated stimulus sets that involved other pairs of prepositions. This left us with a total of 10 sets of stimuli, and 30 total sketches (3 variants in each set). The full set of final sketches used in this experiment, including all three variants, can be found in Appendix A. For retrieval we used Forbus *et al.*'s (1994) MAC/FAC model of similarity-based reminding. MAC/FAC's case library consisted of the sketched versions of the stimuli from the original experiments (all three variants for every stimulus).

4.3.2.3 Experimental Design and Results

SpaceCase Experiment 2.1

For the first experiment, SpaceCase Experiment 2.1, every variation of each input sketch was run through the SpaceCase model to determine the applicability of the intended preposition (*in* or *on*) to the initial situation. This is similar to one experiment run against the human subjects to show the applicability of the preposition to each variant of the initial stimuli. The results are summarized in Table 5 below, these results are averaged across all of the stimuli, there were 30 total sketches, 10 in each category.

Table 5. Average applicability of the appropriate preposition (*in* or *on* depending on the stimulus) for each of the stimuli in the experiment as determined by SpaceCase

Initial sketch	0.363
Plus variant	0.859
Minus variant	0.243

These results are consistent with the human-subject trials where the plus variant was given the highest applicability rating, the initial sketch an in-between rating, and the minus variant, the lowest applicability rating. These results also pointed out some weaknesses in our current version of SpaceCase which will be addressed in a later version. For example, one of the stimulus sets involved a block on top of a building (pictured in Figure 18). For this particular stimulus, the rule for the ground acting as a container fired, since a building can be inferred to be a container based on gens inferencing in the knowledge base. Clearly, a building can be a container, but in this particular case (a block on the roof of the building) the support relationship pictured is not one of containment (as opposed to a person inside of the building, which is a containment relationship). SpaceCase needs to be able to use visual properties as well as conceptual properties to ensure that containment is actually occurring in a given scene.

SpaceCase Experiment 2.2

To provide a baseline of comparison, the next experiment, SpaceCase Experiment 2.2, involved probing each case library with the initial variants of the sketches to see which sketch was retrieved. This setup is similar to human-subject trials where there was no spatial language. As expected, the initial sketch was retrieved in all cases.

The experimental design in SpaceCase Experiments 2.2 and 2.3 is slightly different than the original Feist and Gentner design; this is due to the way that MAC/FAC works (for a more in depth

discussion of MAC/FAC, please see Chapter 3). In the original experiment, subjects would be shown only the initial variants during the encoding stage, and then retrieval would be tested using all three variants. For each stimulus shown during the retrieval trials, the subject would indicate whether they had seen that particular stimulus before (during the encoding trials). MAC/FAC operates with a probe case and a case library and returns the object in the case library that it determines to be most similar to the probe. So, in our experiments, all the variants of all of the stimuli are added to the case library, and then an individual sketch (the initial variant in SpaceCase Experiment 2.2 and the initial variant + spatial language in Experiment 2.3) is used as the probe.

SpaceCase Experiment 2.3

For the final experiment, SpaceCase Experiment 2.3, the sketches were run again with the model, but we added the formal equivalent of the spatial language indicated by the sentence into the representation of the probe sketches. This was done by asserting a statement with the spatial preposition information (e.g. “The spider is on the bowl” would be asserted as `(on-Generic Spider Bowl)`) in the case for the probe sketch. Doing this before running MAC/FAC led to retrieving the plus variant of the sketch from the case library rather than the initial sketch in all 10 trials. This is consistent with the human-subjects results that subjects with the spatial language sentence were much more likely to mistakenly recognize the plus variant.

4.4 GEOMETRIC SHAPES EXPERIMENTS

4.4.1 INTRODUCTION

The experiments in the previous section investigated labeling scenes with spatial prepositions, when given likelihood rules indicating how to update beliefs. This section examines a different problem: how to learn the contents of different spatial language categories. That is, given a set of sketches, can

we learn to classify them into categories based on the spatial relationships present and by doing so, learn the important factors necessary to create the categories? We use SEQL (for a detailed description of the SEQL algorithm, please see chapter 3), an existing model of analogical generalization to construct relational descriptions from stimuli that are input as hand-drawn sketches. In this set of experiments we are attempting to learn to distinguish between *in*, *on*, *above*, *below* and *left* after being trained on simple sketches exemplifying each preposition. Again we rely on CogSketch to automatically compute qualitative spatial relationships for each of the sketched stimuli.

4.4.2 GEOMETRIC SHAPES EXPERIMENT 1

4.4.2.1 Input

Input was provided as sketches created using CogSketch. Each sketch contained two geometric shapes named figure/ground and conceptually labeled with their common shape names (for example in the figure below, in the *in* example the square was named “figure” and conceptually labeled “square”). All of the stimuli were constructed from simple shapes from the set {circle, triangle, rectangle, square}. In the first experiment, the library of sketches used contained 50 sketches. Each sketch was designed to be a good example of one of five spatial prepositions: *in*, *on*, *above*, *below* or *left* with 10 sketches created for each of the five prepositions. By “good example” we mean that each sketch would be easily and unequivocally recognized as an example of the English use of that preposition. For example, for all of the *on* sketches, the figure object was smaller than the ground object and the entire bottom surface of the figure object was in contact with the top surface of the ground object. For all of the *in* sketches, the figure object was smaller than the ground object and was completely contained by the ground object which was a closed shape. Each preposition had examples containing different shapes in the figure and ground roles. All sketches were 2-dimensional and were drawn from the side-view

perspective. One sketch for each of the five prepositions is shown in Figure 19. All of the sketches used in the experiment are available in Appendix B.

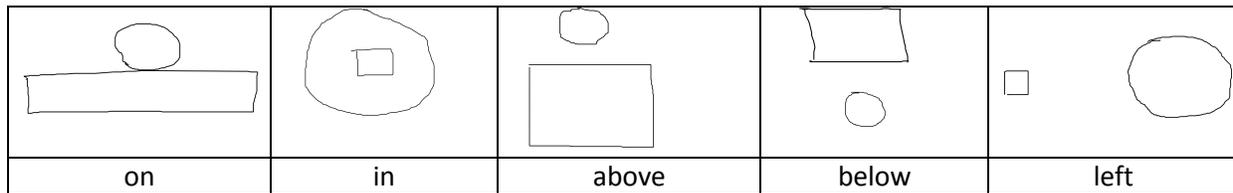


Figure 20. Examples of the sketched inputs for Geometric Shapes Experiment 1.

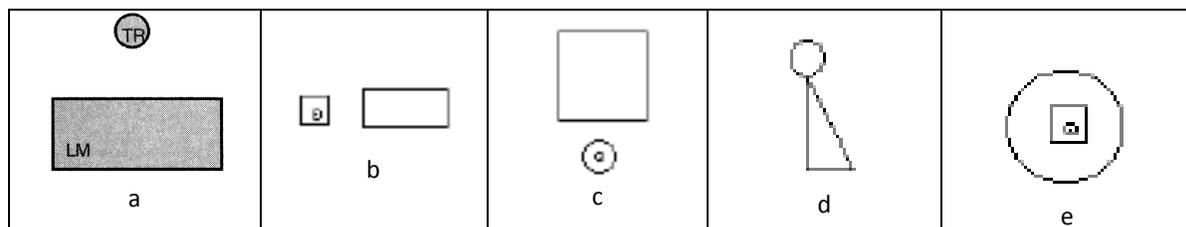


Figure 19. Examples of the stimuli that served as inspiration for the inputs for Geometric Shapes Experiment 1. Figure a taken from (Regier & Carlson, 2001) b-e from (Regier, 1995).

The sketches were inspired by, and drawn from, stimuli in the psychological literature studying spatial preposition use in humans, particularly from studies where the stimuli were simple geometric shapes. The sketches for *above* and *on* were taken in part from examples provided in Regier (1995). Other sketches for *left* and *above* were created based on information from Gapp (1995a, 1995b), whose experiments explored the effect of distance and shape (extent) and size of the ground in judgments of applicability for projective spatial relationships. The sketches were also informed by a variety of experiments that discuss limitations on regions of acceptability for prepositions such as Logan and Sadler (1996) and Regier and Carlson (2001). Figure 20 shows examples of the original stimuli from previous studies that served as inspiration for our input.

Initial processing is done on each sketched stimulus to extract visual information from the ink. This information is meant to approximate high-level visual processing. For example, RCC-8 relations (Cohn, 1996) are computed between the objects in the sketch to determine topological relationships

such as touching (RCC8-EC) and inside (RCC8-nTPP). We use these qualitative spatial relations as one source of perceptually salient relationships in the sketch. To review, CogSketch also automatically computes a variety of other qualitative spatial relationships from the ink. For example, spatial processing identifies groups of glyphs that form connected and contained glyph groups. In the latter case it also specifies which glyph acts as the container and which acts as the insider. Also computed are positional relationships (i.e., *above* and *right-of*) between all pairs of glyphs in a sketch that are disjoint from each other. It is important to note that *above* as computed by CogSketch is very different from its English language counterpart. The spatial relationship *above* in CogSketch is derived by comparing the relative positions of the centers of area of the bounding boxes of the glyphs involved. This alone is not enough information to parse different prepositions. For example, the positional relationship *above* shows up in the generalizations for both *above* and *on*.

Our model does some minimal additional processing based on the spatial relationships computed from the sketch. Each occurrence of a glyph name in a fact was replaced with the label “figure” or “ground” instead of the internal CogSketch reference. For consistency, positional relations are always rewritten so that the figure is in the first argument and the ground is in the second argument i.e. (*above ground figure*) would be rewritten (*below figure ground*). This is necessary because to avoid duplication, CogSketch only computes the above facts. For each sketch, this visual information and any conceptual information about the entities in the sketch is stored as a case. Any unnecessary information, like bookkeeping facts representing specifics of our implementation, is filtered out since we do not view them as psychologically relevant. All filtering and processing procedures were done over the entire case library of exemplars – individual sketches were never singled out for processing. The parameters describing the filtered objects are available in Appendix C.

4.4.2.2 Experimental Design and Results

All 50 sketches were run through SEQL using an assimilation threshold of 0.9. Our goal in doing these experiments is to see whether we can achieve human-like classification results automatically and what specific set of factors are needed to do so. The fifty simple, unlabeled sketches were automatically classified by SEQL into the five generalizations expected (corresponding to *in*, *on*, *above*, *below* and *left*) based on the combination of perceptual features in each case. These unsupervised learning results were stable over a variety of match threshold values between 0.8 and 0.9. Inspection of the generalizations created shows the common features of the sketches that created that generalization. Figure 21 shows the generalization that was created for *on*, all of the facts appeared in all member sketches, so there are no probabilities, all are definite facts. All five of the generalizations are available in Appendix D. Figure 22 shows the process used to create the spatial categories from the sketched stimuli.

(enclosesHorizontally ground figure) (connectedGlyphGroupTangentialConnection figure ground) (connectedGlyphGroupTnagentialConnection ground figure) (rcc8-EC figure ground) (above figure ground)
--

Figure 21. The generalization created for the preposition *in*

The information included in the generalization is visual information based on the spatial arrangement of the glyphs in the sketch. Looking at the facts generalized, it makes sense that the salient perceptual information needed to assign the label *on* would be a combination of tangential connection between the figure and the ground and the figure being above the ground (keeping in mind that all of the input sketches were canonical examples of *on* with the full bottom surface of the figure being in contact with the top surface of the ground). It is also important to note again that the simple, individual qualitative relations computed by CogSketch are not enough to generate these categories, it is their combination that leads to meaningful results.

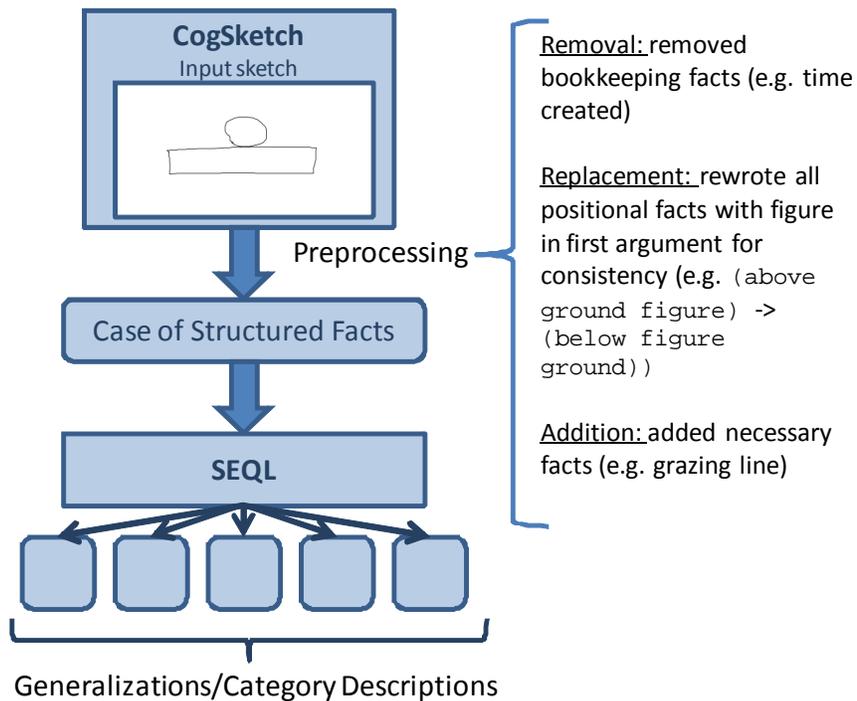


Figure 22. Experimental design for Geometric Shapes Experiment 1

These are surprisingly good results considering that we used only 10 sketches for each preposition. Also, relatively few facts were needed in each case to determine which category a sketch fell into. The average number of facts per generalization was 5.6. The largest number of facts in a generalization was 7. It is also important to note that not just any set of facts will result in a useful classification. If bookkeeping information is not filtered out, it will overwhelm the cases and the categories that result will be meaningless. Also, object-centric perceptual information (such as relative size and roundness) had to be filtered out, as it ended up being irrelevant to the spatial preposition categories and was adding noise to the similarity comparisons.

Likewise, while doing these experiments, we found several additional spatial relationships that had not previously been computed that were needed to create meaningful generalizations. In order to get the *above* and *below* cases to generalize correctly, we added a computation about the grazing line.

The grazing line is a horizontal line that grazes (is tangential to) the very top of the ground object. Regier and Carlson (2001) suggest that *above* ratings from human subjects were sensitive to the grazing line and we found the same for our system in our experiments. Whenever the figure and ground objects do not touch (RCC8-DC) a grazing line computation is triggered and one of two facts `{{(aboveGrazingLine figure ground), (belowGrazingLine figure ground)}}` is added to the sketch case.

When glyphs partially overlap, a fact is also asserted based on percentage of total area overlap (`LessThan10Overlap`, `DefiniteOverlap`, `MoreThan90Overlap`). This computation is a rough approximation based on the blob boundaries of the two glyphs. These facts are useful for disambiguating cases of partial overlap from those that are just poorly drawn examples of *in* or *on* and are computed for every sketch where overlap exists. Since none of the simple sketches in Geometric Shapes Experiment 1 had any overlap, these facts do not show up in any of the generalizations created, however they are important in Geometric Shapes Experiment 2 described in the next section.

Table 6. Summary of the perceptual relationships that form the content of the generalizations created in Simple Geometry Experiment 1 and the categories in which they appear

Relationship	Categories
Horizontal enclosure	<i>below, above, on</i>
Vertical enclosure	<i>left</i>
Left of	<i>left</i>
RCC8-DC (disjoint)	<i>below, above, left</i>
Above	<i>above, on</i>
Below	<i>below</i>
Above grazing line	<i>above</i>
Below grazing line	<i>below</i>
Contained glyph group	<i>in</i>
RCC8-NTPP/TPP (inside)	<i>in</i>
Connected glyph group	<i>on</i>
RCC8-EC (touching)	<i>on</i>

The set of facts retained in generalizations is summarized in Table 6 along with the categories they appear in. It is interesting that this small set of relationships is sufficient to distinguish between these prepositions. Efforts were made to remove redundant and unnecessary information. For example, in addition to recognizing contained glyph groups, CogSketch also asserts information about which object is designated as the container and which is the insider. At this level of classification, removing that information had no impact on the generalizations created. Keeping just the information that the ground and the figure form a contained glyph group is enough to ensure that the correct generalization will form.

4.4.3 GEOMETRIC SHAPES EXPERIMENT 2

In Geometric Shapes Experiment 1, all of our stimuli were very good examples of the type of preposition they were meant to represent. In the second experiment, we wanted to look at what would happen if we added stimuli that were “less good” examples of the same prepositions. For example, if

instead of having some stimuli where the figure was strictly above the ground and some stimuli where the figure is strictly to the left of the ground, what happens if there are some stimuli where the figure is both slightly to the left of and slightly above the ground? Stimuli like these are often used in the psychological literature to test the boundaries of different prepositions. Subjects will be shown a variety of ambiguous arrangements and be asked either (1) which preposition best describes the scene or (2) to provide a goodness judgment for how well a given preposition fits the scene. Since these stimuli are, by definition, outliers, they were not used in Geometric Shapes Experiment 1 where the goal was to examine the core constituents of preposition categories. Instead, in this experiment, the goal is to see how the ambiguous sketches are assimilated (or not) into existing canonical categories.

4.4.3.1 Input

The input for Geometric Shapes Experiment 2 was very similar to that for Geometric Shapes Experiment 1. The same 50 sketches from Geometric Shapes Experiment 1 were used. In addition, 20 new sketches were created which were more complicated (non-standard) and/or ambiguous cases of the spatial prepositions used, the full set of these sketches is in Appendix F. Figure 23 below shows four sketches from the 20 added that illustrate different reasons for inclusion. The sketch in part a is an example of an ambiguous situation (the circle could be *on* or *in* the square). The sketch in part c is a non-standard instance of *on* with vertical as opposed to horizontal support (this is similar to the case “the picture is *on* the wall”). For the rest of this discussion, the 50 original sketches from Geometric Shapes Experiment 1 will be referred to as the simple sketches and the 20 additional sketches added in Geometric Shapes Experiment 2 will be referred to as the complex sketches.

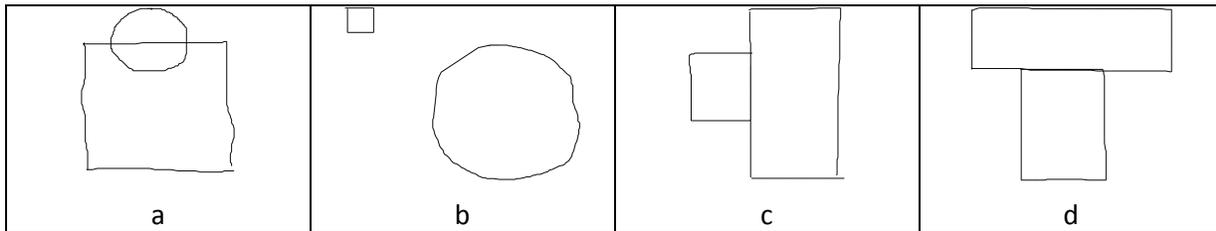


Figure 23. Examples of the ambiguous stimuli used in Geometric Shapes Experiment 2, highlighting the types of variations included

The 20 complex sketches obviously could not cover every possible arrangement of figure and ground, so we focused on the following deviations:

- sketches where the figure overlaps the ground by varying amounts (ambiguous between *in* and *on*) as in Figure 23, part a.
- sketches ambiguous between *above* and *left* (Figure 23, part b)
- sketches where the figure is attached to the side of the ground – vertical as opposed to horizontal support (as in Figure 23, part c) or where the ground is sloped
- *on* and *above* examples where the figure was larger (larger vertical or horizontal extent) than the ground (Figure 23, part d)

The idea that some scenes are better examples of certain prepositions than others is common in the literature. For example, Logan and Sadler (1996) argue that for spatial templates, there are three regions of acceptability for spatial relationships: the good region, the region of examples that are not good, but are acceptable and the region of unacceptable examples. The 20 complex sketches are intended to fall into the acceptable but not good category.

4.4.3.2 Experimental Design and Results

As in Geometric Shapes Experiment 1, we use CogSketch to create a case of facts from each sketch and then use SEQL to classify the sketch cases. In this set of experiments the simple sketches were classified using SEQL in a first-pass at classification which created the same generalizations as in

the first experiment. Once the base generalizations based on the simple sketches were in place, the complex sketches were added and the SEQL generalization algorithm was run again to incorporate the new sketches. Several different runs were done with varying match threshold values, and again we found good results at the 0.8 and 0.9 levels.

As mentioned in the previous section, the 50 simple sketches created five generalizations, one corresponding to each of the prepositions (*on*, *in*, *above*, *below* and *left*). This result was unchanged in this experiment. After the complex sketches were added, the generalizations were changed in the following way:

- The ambiguous *above/left* sketches divided – the one that was most like the *above* sketches joined that generalization while the other two created a separate generalization.
- The sketches where the figure overlapped the ground by varying amounts all joined the *in* generalization.
- The *on* category assimilated all of the other sketches that were meant as complex or ambiguous examples of that preposition.
- There was one example of an ambiguous *over* sketch that was not incorporated into any of the generalizations and remained an exemplar.

The incorporation of these instances into the overall generalization altered the facts that were considered part of the generalization as can be seen in Figure 24 and Figure 25.

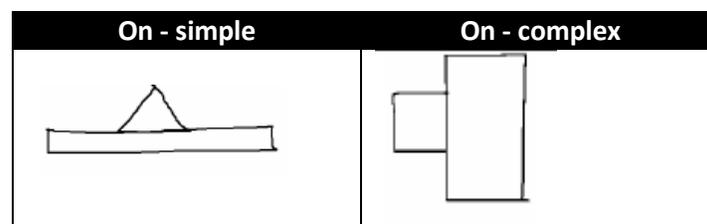


Figure 24. An example of one of the original examples of *on* and one of the *complex* examples

```

--DEFINITE FACTS:
(rcc8-EC figure ground)
(connectedGlyphGroupTangentialConnection ground figure nil)
(connectedGlyphGroupTangentialConnection figure ground nil)
(ConnectedGroup figure ground)
--POSSIBLE FACTS:
88%: (above figure ground)
65%: (enclosesHorizontally ground figure)
--UNLIKELY FACTS:
6%: (enclosesHorizontally figure ground)
6%: (enclosesHorizontally figure ground)
6%: (enclosesVertically ground figure)
6%: (leftOf figure ground)
6%: (rightOf figure ground)

```

Figure 25. The new generalization for *on* after the complex sketches have been added

Clearly this new generalization covers a much broader arrangement of objects. However, it is important to note that all sketches that were included in this generalization depict a relationship that could reasonably be classified in English using the preposition *on*. The facts common to all of the sketches were those that pertain to connectedness/support which is key to the use of the preposition *on*. The other facts allow for some variation in exactly how the two objects are connected in space and how the support is provided to the figure by the ground. This is the general pattern for how the existing generalizations were altered by the addition of the complex sketches – the core components of the preposition (containment, support, positioning relative to the grazing line) remained largely intact with small variations in things like vertical/horizontal containment, left/right positioning, and amount of overlap. The *in* generalization changed to allow for either contained groups or significant overlap to signal containment. *Over* retained the requirement that all instances have the figure above the grazing line of the ground, but allowed for variation in the horizontal containment constraint. The two ambiguous *left/above* instances that formed their own generalization both failed the grazing line test.

The Geometric Shapes experiments provide insight into the core components of a scene that

might signal the use of a spatial preposition to describe it. They also suggest how core spatial categories may be stretched to cover non-standard cases. However, there were several shortcomings in these experiments. First of all, the use of geometric shapes limits the usefulness of the categories created since much of spatial preposition use can be attributed to the functional features of the objects involved. Secondly, these experiments were done with unsupervised learning (i.e. unlabeled inputs). This is very unlike the way that people are exposed to prepositions where arrangements of objects are most likely labeled verbally. Finally, these experiments examined only English prepositions. These shortcomings are all addressed in the next set of experiments involving the containment and support relations in English and Dutch.

4.5 CROSS LINGUISTIC EXPERIMENTS: CONTAINMENT-SUPPORT RELATIONS IN ENGLISH AND IN DUTCH

4.5.1 INTRODUCTION

In the Geometric Shapes experiments I looked at automatically creating categories of spatial language based on sketched inputs. In this set of experiments I set out to correct several shortcomings of the previous work. First of all, in the previous experiments, the input sketches were unlabeled and I relied on pre-processing to prune all irrelevant facts. While this worked in the model, it seems likely that it is not very similar to the way that humans learn spatial categories. Instead, humans hear spatial scenes in their lives described (labeled) from a very young age (e.g. “put the toys in the box”) and the hearers learn over time what factors of a scene are needed to make judgments and which can be discarded. So, I wanted to repeat the experiments with labeled training data and to decrease the amount of preprocessing that needed to be done.

The previous experiments also dealt exclusively with scenes composed of simple geometric shapes. This was useful for allowing us to focus on the geometric aspects of scenes that contribute to spatial language use, but did not allow us to look at the range of functional aspects of scenes that have also been shown to influence spatial language use. In this set of experiments I wanted to focus on scenes involving real-world objects so that functional properties could be considered in category formation as well as geometric ones.

The final aspect of spatial preposition use that I was hoping to address with this set of experiments is cross-linguistic variation. Given that spatial language varies so much from culture to culture, I wanted to specifically see if my system could learn the labels for the same set of sketches in more than one language using exactly the same inputs and the same classification method – but with different, language-specific labels.

Luckily I was able to address all three of these goals (1) labeled training sets (2) input with real world objects and (3) cross-linguistic variation in one set of experiments. This section describes my work modeling the results of a Gentner and Bowerman study looking at the containment and support relationships in both Dutch and English.

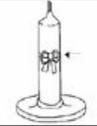
4.5.2 CROSS-LINGUISTIC EXPERIMENT

4.5.2.1 Original Gentner and Bowerman Study

As previously mentioned, this work is based on a set of psychology experiments conducted by Dedre Gentner and Melissa Bowerman (2009). Gentner and Bowerman were testing the *Typological Prevalence hypothesis*, that the frequency with which distinctions and categories are found across the world's languages provides a clue to conceptual “naturalness” and how easy that particular distinction is to learn. To explore this, they focused on a subset of spatial prepositions in English and Dutch. The

English and Dutch languages divide the support-containment continuum quite differently. In English there are two prepositions: *in* is used for containment relationships and *on* is used for support relationships. However, Dutch distinguishes three different forms of support. The prepositions for Dutch and English are outlined in Table 7.

Table 7. The containment and support spatial prepositions in English and Dutch

English	Dutch	Relationship	Example
on	op	Support from below	
on	aan	Hanging attachment	
on	om	Encirclement with contact	
in	in	Containment	

Bowerman and Pederson found in a previous study (1992) that some ways of dividing up the containment-support continuum are very common cross-linguistically while others are relatively rare. English follows a more linguistically common approach by grouping all support relations together into the *on* category while the Dutch *op-om-aan* distinction is extremely rare. Both use the very common *in* containment category. Following the Typological Prevalence Hypothesis, both English and Dutch children should learn the common and shared category of *in* around the same time. It should take Dutch children longer to learn the rare *aan/op/om* distinctions for support than it takes the English children to learn the common *on* category.

Gentner and Bowerman tested children in five age groups (2, 3, 4, 5, and 6 years old) as well as adults who were native speakers of English and Dutch. Each subject was shown a particular arrangement of objects and asked to describe the relationship in their native language. In the original experiment, 3-dimensional objects were used. So, for example, a subject would be shown a mirror *on* the wall of a doll house and asked “Where is the mirror”. The set of all stimuli is shown in Table 8 below.

Table 8. The original stimuli from the Gentner and Bowerman experiment

<i>op/on</i>	<i>aan/on</i>	<i>om/on</i>	<i>in/in</i>
cookie on plate	mirror on wall	necklace on neck	cookie in bowl
toy dog on book	purse on hook	rubber band on can	candle in bottle
bandaid on leg	clothes on line	bandana on head	marble in water
raindrops on window	lamp on ceiling	hoop around doll	stick in straw
sticker on cupboard	handle on pan	ring on pencil	apple in ring
lid on jar	string on balloon	tube on stick	flower in book
top on tube	knob on door	wrapper on gum	Cup in tube
freckles on face	button on jacket	ribbon on candle	Hole in towel

The results of the study were consistent with the Typological Prevalence hypothesis. Specifically, Dutch children are slower to acquire the *op*, *aan*, *om* system of support relations than English children are to learn the single *on* category. Both groups of children learned the *in* category early and did not differ in their proficiency using the term. Across all prepositions, English-speaking 3 to 4 year old children used the correct preposition 77% of the time, while the Dutch children used the

correct preposition 43% of the time. Within the Dutch children, the more typical *op* category was learned sooner than the rarer *aan* and *om* categories. For a more detailed description of the results, please see the original paper (Gentner and Bowerman, 2008).

4.5.2.2 Materials

All 32 original stimuli from the Gentner and Bowerman study were sketched using CogSketch. Each sketch was stored as a case containing: (1) the automatically computed qualitative spatial relationships and (2) information about the types of objects in the sketch. In the original experiment subjects were cued as to which object should be the figure (e.g. “where is the mirror”) and which should be the ground. To approximate this, each sketch contained two glyphs, one named figure and one named ground, and these names were used by the model. Recall that names in CogSketch are just strings that are used to refer to the objects. Each object was also conceptually labeled using concepts from the KB. For instance, in the mirror *on* the wall stimulus, the mirror was declared to be an instance of the concept `MIRROR` and the wall was labeled as an instance of `WallInAConstruction`. All of the sketches used can be found in Appendix G and Appendix H lists the concepts used in conceptual labeling.

When people learn to identify spatial language categories in their native languages, they learn to focus on the relationships between objects, and to retain only the important features of the objects themselves rather than focusing on the surface features of the objects. This allows people to correctly use prepositions to describe a relationship even if they have not seen the figure or ground object before, or if they have seen the same objects in different configurations. For example a bandaid *on* a leg as opposed to a bandaid *in* a box. The preposition depends not on the fact that there is a bandaid involved, but on the specific properties of the bandaid that are applicable in each situation. As noted above, having conceptual labels and a knowledge base allows us to simulate this type of knowledge. For

each conceptual label, additional concepts from its genls hierarchy were extracted from ResearchCyc. The genls hierarchy specifies subclass/superclass relationships between all the concepts of the KB. So, for example, *Animal* and *Dog* would both be genls of *Daschund*. Here we were particularly interested in facts relating to whether objects were surfaces or containers – since this has been shown to be an important factor in spatial preposition use (e.g. Feist and Gentner, 2003).

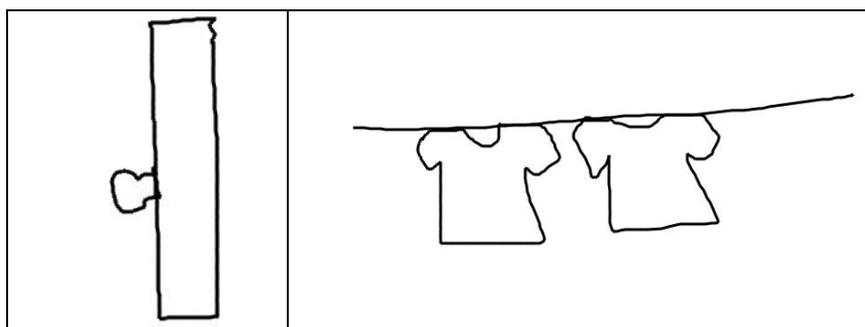


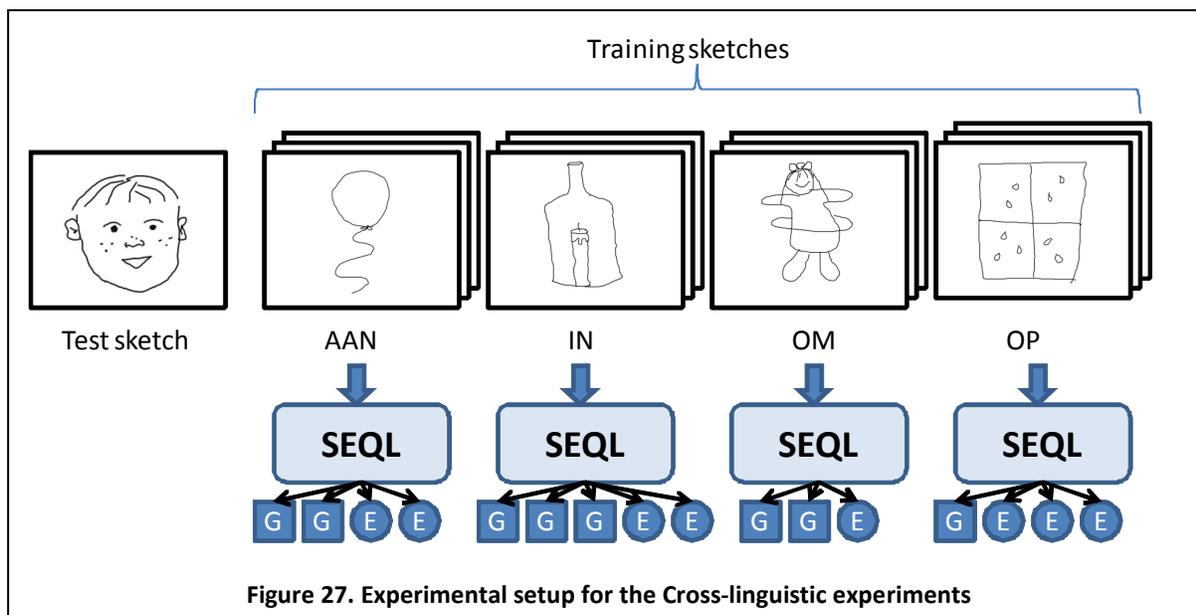
Figure 26. Two examples of *aan* drawn to highlight the type of connection.

In the original study, the physical objects used as stimuli were manipulated to make the important relationships more salient to subjects. We approximated this by drawing our sketches so as to highlight the important relationships for the individual spatial language categories. For example, the sketches for *aan* that required showing a connection by fixed points were drawn from an angle that made the connectivity between the parts observable. Figure 26 below shows two *aan* sketches: knob *aan* door and clothes *aan* line. They are drawn from perspectives that allow the system easy access to the point-contact relationship.

The basic spatial category learning algorithm is this: For each word to be learned, a generalization context is created. Each stimulus representing an example of that word in use is added to the appropriate generalization contexts using SEQL. (Since we are looking at both Dutch and English, each example will be added to two generalization contexts, one for the appropriate word in each language.) Recall that SEQL can construct more than one generalization, and can include unassimilated

examples in its representation of a category. Each generalization context acts as a defacto label, making this set of experiments a type of supervised learning.

To test this model, we did a series of trials constructed as shown in Figure 27. Each trial consisted of selecting one stimulus as the test probe, and using the rest of the sketches as the training examples to create the categories. The test probe (T) was then labeled as follows: We let the score of a generalization context be the maximum score obtained by using SME⁴ to compare T to all of the generalizations (G) and unassimilated examples (E) in that context. The word associated with the highest-scoring generalization context represents the model's decision. The trial was correct if the model generated the intended label for that stimulus. There were a total of 32 trials in English (8 for *in* and 24 for *on*) and 32 trials in Dutch (8 each for *in*, *op*, *aan*, and *om*) one for each stimulus sketch.



⁴ See Chapter 3 for a discussion of the SME algorithm

4.5.2.3 Results

The results of our experiment are shown below. The generalizations and numbers given are for running SEQL on all the sketches for a category. Table 9 below summarizes the number of sketches that were classified correctly, for each preposition the number is out of 8 total sketches except for English *on* which has 24 total sketches. All results are significantly different from chance ($p < 10^{-4}$), except for the English *in* ($p < 0.2$). For an in-depth discussion of the error patterns, see the Error Analysis section.

Table 9. Results for both English and Dutch. All are significant, except for English *in*

English			Dutch		
<i>in</i>	6	75%	<i>in</i>	6	75%
			<i>op</i>	7	87%
<i>on</i>	21	87%	<i>aan</i>	6	75%
			<i>om</i>	8	100%

Recall that within each generalization context, SEQL was free to make as many generalizations as it liked. SEQL was also able to keep some cases as exemplars if they did not match any of the other cases in the context. The table below summarizes the number of generalizations and exemplars for each context for one run of the experiment.

Table 10. Number of generalizations and exemplars created within each context

	English		Dutch			
	<i>in</i>	<i>on</i>	<i>in</i>	<i>op</i>	<i>aan</i>	<i>om</i>
Generalizations	2	6	2	2	3	3
Exemplars	2	0	2	2	0	2

At first the amount of variation within the contexts might seem surprising. However, since the stimuli were chosen to cover the full range of situations for each context it makes more sense.

Consider the Dutch category *op*. The 8 sketches for this one generalization included very different situations: clingy attachment (e.g. *sticker op cupboard*), traditional full support (e.g. *cookie on plate*) and covering relationships (e.g. *top on jar*). Two of the English generalizations are shown in Figure 28 and Figure 29. For each generalization the cases that were combined are listed followed by the facts and associated probabilities.

Best Generalization IN

Size: 3

(candle in bottle, cookie in bowl, marble in water)

--DEFINITE FACTS:

(rcc8-TPP figure ground)

--POSSIBLE FACTS:

33%: (Basin ground)

33%: (Bowl-Generic ground)

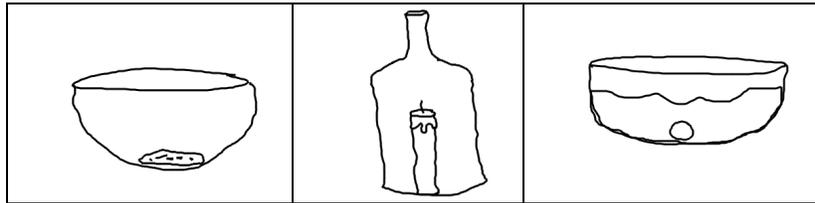


Figure 28. One generalization created for in (which is the same in both Dutch and English) and the sketches that were the generalized cases

Best Generalization ON

Size: 2

(top on tube, lid on jar)

--DEFINITE FACTS:

(Covering-Object figure)

(above figure ground)

--POSSIBLE FACTS:

50%: (definiteOverlapCase figure ground)

50%: (rcc8-PO figure ground)

50%: (rcc8-EC figure ground)

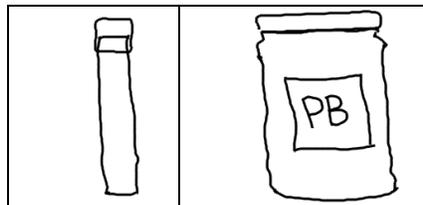


Figure 29. One generalization created for English on and the sketches that were the generalized cases

4.5.2.4 Error Analysis

Closer examination of the specific errors made by SEQL is also illuminating. For example, both the Dutch and English experiments failed on two *in* stimuli. It was the same two stimuli for both languages: flower *in* book, and hole *in* towel. The first case, flower *in* book, is hard to represent in a sketch. In the original study, actual objects were used, making it easier to place the flower in the book. It is not surprising that this case failed given that it was an exemplar in both *in* contexts and did not share much structure with other stimuli in that context. Hole *in* towel fails for a different reason. The ResearchCyc knowledge base does not have any concept of a hole. The closest suggested category is `Cavity` which is a type of `ConcaveTangibleObject` which would be useful for describing a hole in the ground (which might be considered a type of container) but not for a hole in a towel (which is just the absence of material). Moreover, how holes should be considered in spatial relationships seems different than for physical objects.

Many of our errors stem from the small size of our stimuli set. For contexts that contained multiple variations, there were often only one or two samples of each. An interesting future study will be to see how many stimuli are needed to minimize error rates. (Even human adults are not 100% correct on these tasks.) Interestingly, *om* is one of the prepositions that is harder for Dutch children to learn (it covers situations of encirclement with support). However, it was the only Dutch preposition for which our system scored 100%. This again is probably explainable by sample size. Since the entire context contained only cases of encirclement with support, there was more in common between all of the examples as opposed to categories with more variation where the individual sketches had less in common.

4.5.2.5 Discussion

Our model was able to successfully learn the support-containment prepositions in both Dutch and English with a small number of training trials. We see three lines of investigation suggested by these results. First, we would like to expand our experiments to include more relationships (e.g. under, over, etc). Second, we would like to expand to other languages. For example, Korean uniquely divides the containment relationship into tight fit and loose fit relations. Third, we are in the process of building a sketch library of more instances of spatial relations. With more sketches, we will have additional evidence concerning the coverage of our model.

There is also clearly a tradeoff between using a cognitively plausible number of training examples and having enough training examples to get good generality. For example, adding the ability to automatically extract the important object types and features (e.g. containers) and ignore the spurious ones (e.g. that something is edible) requires enough training examples to be able to extract the necessary patterns. We are planning future experiments to examine this issue by varying the number of training trials used. It will also be interesting to see if we can use the same set of experiments to model the development of spatial language categories in children by varying the availability of different types of information. Experiments like the Gentner and Bowerman work modeled here often report on common misuses of prepositions by children (i.e. over extension of common prepositions). It would be interesting to try to replicate these types of errors by varying the content of the cases used in generalization and the assimilation threshold.

4.6 RELATED WORK

All of the experiments described in this chapter are attempts to model different aspects of spatial preposition usage. This is a rich area of research in the cognitive modeling community, and there are a variety of approaches, each with their own benefits and tradeoffs.

A number of models of spatial preposition usage rely on representational templates that are created by hand. For example, Herskovits (1980, 1986) categorizes spatial language into use cases based on object and contextual features as well as typicality. Along the same lines, Logan and Sadler (1996) classify geometric scenes using spatial templates. Template based models like these require *a priori* an exhaustive list of the use cases/templates available, mechanisms for selecting the correct one, and an account of what modifications can be made to fit an imperfect template to a scene. By contrast, our use of SEQL produces relational templates automatically and reduces the imperfect fit problem to structural alignment. The computational model of DiTomso *et al* (1998) attempts to computationally implement a model based on ideal meanings. Their model incrementally refines its understanding of a preposition, starting with a set of all ideal meanings and then eliminates those that do not fit with the characteristics of the objects filling the figure and ground roles. Their model is interesting in that it has the ability to rule out unlikely scenarios (e.g. the box is *in* the book vs. the book is *in* the box) however, it suffers from the same drawbacks as other template-based models –the templates must be hand constructed, and it is virtually impossible to build an exhaustive set of templates.

A similar approach is to use functions to delineate areas of applicability for different prepositions. This is the approach taken by the VITRA project, in particular the SOCCER application (Andre, Herzog & Rist, 1988; Blocher & Stopp, 1998), which aims to create radio-style reports of soccer games. In this system, cubic spline functions are created for every combination of reference (ground) object and preposition which describes the degree of applicability with respect to the position of a

located (figure) object. This is made easier by the assumption that the background is always a soccer field, making the potential landmarks constant (goals, penalty lines, corners, etc). So for example, there would be one function representing the acceptable classification for “*in front of the left goal*”, another for “*left of the left goal*”, etc. There would be another set of functions for the right goal, each corner, and every other discernable landmark on the soccer field. There are also more general areas defined such as “field”, “right-half of field”, etc. This approach allows SOCCER to describe the location of any located object at various levels of detail and in real time. However, this approach is limited to the domain at hand and other domains where the background is known *a priori* as are the potential relations and the spline functions. Fuhr, *et al.* (1998) have a system that can describe the location of building toys in 3D space to a human partner (and react to the human partner’s locative descriptions). They also use a set of functions to define regions of acceptability. In this case each object partitions the 3D space into its own set of 3D acceptance volumes, in the process creating 79 acceptance volumes for each object. When an object is moved, its acceptance volumes must be updated.

There have been many attempts to create models of spatial prepositions. However, most of these have been based, like our first set of classification experiments, on purely geometric models (Logan & Sadler, 1996; Regier, 1996; Gapp, 1995). Edwards and Moulin (1998) use the Voronoi model of space (Okabe, Boots & Sugihara, 1992) to computationally model spatial relations, but again, that model refers only to the topology of a scene. One model that is often cited is Terry Regier’s (1995, 1996) model which predicts spatial term acceptability judgments in a variety of languages based on five-frame movies. Regier and Carlson’s (2001) Attentional Vector Sum (AVS) model uses an attentional beam that is focused on the landmark (ground) at the point that is closest to being aligned with the located object (figure). Attentional strength is maximized at the focus of the beam and drops off with distance. A vector is also projected from each point on the landmark towards the located object. These two

components are joined to form an *attentionally weighted vector sum*. Each vector is multiplied by the attentional strength at its root and these values are summed together. This vector sum represents the overall spatial alignment between the two objects. They are then measured with respect to the reference axis for prepositions (for example the upright vertical axis for above). This model was consistent with human results when run with simple shape drawings where a small circle was placed at various locations “over” larger triangles, squares, and “L” shapes (Regier & Carlson, 2001). While these results were encouraging, they are quite limited in scope for the same reason as our geometric classification experiments, as they only apply to very simple 2-D shapes. Additionally, our model has the advantage of requiring far fewer exposures to fewer total stimuli in order to create spatial language categories.

Recently, there has also been a group of models that attempt to incorporate extra-geometrical/functional attributes of scenes in spatial prepositions (Regier, Carlson, & Corrigan, 2004; Coventry, *et al.*, 2005). The Reiger, Carlson, & Corrigan work is an adaptation of AVS created to be consistent with the findings of Carlson-Radvansky, *et al.* (1999) that showed that the functional parts of objects were important in preposition use. In the new version of AVS, attention is focused on the functional parts of objects using an importance parameter for different regions of objects that is set by the human-user of the model. For example, when modeling a tube of toothpaste *over* a toothbrush, the bristles on the brush are tagged as the functionally important part of the object. This is very interesting work, but requires human intervention to assign the weights to different parts of the objects. Also, in different arrangements, the part of an object that is functionally important may change, meaning that in every new scene the functionality parameters would have to be adjusted. For example, consider the difference in attention to parts of the table in “the table is *on* the rug” and “the apple is *on* the table”. However, our model, while it does consider functionality, is not currently able to separately

consider the functionality of different parts of one glyph. The Coventry *et al.* work uses a connectionist model to predict and simulate the knowledge needed for the dynamic-kinematic routines in the functional geometric framework. This model uses a neural network whose input is descriptions of visual scenes. These descriptions are created using variables to encode various factors that were found to influence *over/under/below/above* judgments in human subjects: orientation, function, appropriateness, and object type (Coventry, Prat-Sala & Richards, 2001).

The main benefit of our approach is the flexibility and extendibility of the system. Since the input is sketches, it is very quick and easy to create more stimuli and to test more arrangements of objects. It also reduces the slippery slope inherent in hand-coded representations. Since conceptual labeling allows us to tie objects in our sketches to concepts in an underlying off-the-shelf knowledge base, functional information can be added through inference, as it was the English/Dutch experiments.

4.7 4.7 GENERAL DISCUSSION

In this chapter we described three sets of experiments looking at the formation and usage of spatial language categories. The first experiments involved the SpaceCase model of spatial preposition use. SpaceCase was able to correctly label occurrences of *in* and *on* in sketched scenes based on evidential rules in a Bayesian updating algorithm. The evidential rules were hand-coded based on the findings of Feist and Gentner (2003). This same model was used in a second set of experiments to model the effect of spatial language on memory encoding by simulating the results of another Feist and Gentner study (2001).

Work with SpaceCase led us to further explore whether the features for category membership could be learned automatically (rather than be coded by hand into the evidentiary rules). This led to the Geometric Shapes experiments, attempting to automatically learn spatial language categories for *in*, *on*, *above*, *below* and *left*. The stimuli in these experiments were simple scenes involving geometric shapes.

By carefully tuning the qualitative spatial relationships available in the exemplar cases, we were able to use SEQL to automatically learn all five preposition categories without supervision.

While SEQL was successful in modeling spatial language with geometric sketches, these experiments did not address the fact that language learning is typically supervised. So, in the third set of experiments we (1) used labeled training data to more closely emulate the way that humans learn spatial prepositions. (2) we used stimuli that contained real-world objects and harnessed the power of the underlying KB to infer the functional information needed in spatial language categories. (3) we used the same set of stimulus sketches, based on a psychology study, and the same general method of classification to learn the containment-support relations in both English and Dutch.

In all of these experiments, we ignore to a large degree the actual geometry of the figure and ground objects, using rough approximations (bound box, blob boundary). Herskovits (1998) argues that often features of the figure/ground are necessary and at other times specific types of schematization are important. For example, many times a solid object acting as a figure can be represented as a point and a path can be conceptualized as a ribbon. Using a bounding box or blob boundary is one type of schematization that in some situations may be indistinguishable from the schematization itself. However examining the objects and their roles in more detail is an interesting area for future work. Others have pointed out that parts of objects often come into play, for example, the top of a table is particularly important when considering the applicability of *on* or the accessible side of a chair when considering the applicability of *in front of*. Again, this is something that we are not currently utilizing as we are dealing with glyphs as atomic units and not further breaking them down into surfaces. However, the ability to segment a glyph is built into CogSketch, and this is another possible direction for future work.

We also are largely ignoring issues of reference frame assignment and ground selection which both influence preposition use. We rely on CogSketch's genre and pose to provide the frame of reference, and all of our stimuli are created with this in mind. In all of the experiments here, the figure and ground objects are unambiguously labeled as such and all of the sketches have only two objects, making ground selection unnecessary. However, both frame of reference and ground object selection are interesting problems for future work, especially on sketches that involve more than two objects and are from varying scales and perspectives.

These experiments demonstrate that sequential generalization combined with sketched input is a powerful method for modeling the learning of spatial language categories. Sketching for input allows us to quickly and easily create any number of inputs containing any number of objects without the biases that arise from hand-coded inputs. Sketching with conceptual labeling also gives us the flexibility to extract functional information about the objects in the sketch from the underlying knowledge base in addition to extracting automatically computed qualitative spatial relationships from the ink.

SEQL with generalization contexts lets us create multiple generalizations within each context, allowing us to capture the variety of concepts often represented by a single spatial language label. Unlike many other models of spatial preposition use, SEQL requires very few training examples to work successfully, although the optimal number of examples is an open empirical question. SEQL is also flexible enough to model multiple languages using the same set of sketches and the same progressive alignment algorithm.

5 MULTIMODAL KNOWLEDGE CAPTURE

5.1 INTRODUCTION

Researchers from psychology and learning sciences have examined the question of whether, and under what conditions, people learn better from multimodal presentations of information than from single-modality information (e.g. Mayer, 2001; Hegarty and Just, 1993). Many of these studies have shown that subjects are able to perform better on tests of retention and transfer when they were presented with multimodal information sources, such as animations with narration or text with diagrams. Indeed many traditional sources of instructional material contain multiple modalities, e.g., textbooks contain both text and diagrams. To exploit such materials, knowledge capture systems should be able to integrate information across modalities into coherent chunks of knowledge.

There are multiple theories as to how and why multimodal sources of information lead to better recall and transfer performance. The *multimedia learning theory* (Mayer, 2001) posits that instead of passively absorbing information, learners cognitively engage with it in an active attempt to understand (see chapter 2 for a more thorough discussion). Under this theory, multimedia presentations of information lead to better understanding because learners actively engage in sense making activities as they attempt to integrate information from the two modalities, and it is this active engagement with the material that leads to deeper learning.

This chapter presents a multimodal knowledge capture system (MMKCap) based on this theory, which asserts that such learning is a five step process:

- 1) Selecting relevant words for processing in verbal working memory
- 2) Selecting relevant images for processing in visual working memory
- 3) Organizing selected words into verbal mental model

- 4) Organizing selected images into visual mental model
- 5) Integrating verbal and visual representations along with existing knowledge.

In MMKCap steps 1 and 2 (selection) are done manually by dividing the text and diagrams into discrete chunks. Text chunks are determined by paragraph structure in the text and by diagram references in the text. Each diagram is its own diagram chunk. Step 3, developing a representation from the text, is done using the EA NLU natural language understanding system. Extraction of information from the diagrams (step 4) is accomplished via the CogSketch sketch understanding system. The final step, Integration, uses the Structure Mapping Engine model of analogy and similarity to perform the cognitive task of comparing and integrating the two representations. Descriptions of these systems can be found in Chapter 3.

After describing each of these steps in more detail, this chapter presents an experiment in which MMKCap is used to learn a chapter of content from a physics textbook: *Basic Machines* (1994). The system is evaluated on its ability to answer the homework questions provided by the publisher. After the evaluation, the related problem of conceptual segmentation is discussed along with a proof-of-concept diagram segmentation system. We conclude with related work.

5.2 MATERIALS

The text chosen to demonstrate the MMKCAP system is *Basic Machines*, a Naval training manual that covers basic physics concepts. *Basic Machines* was chosen due to the relatively high volume of diagrams in the text, and because portions of it have been used in cognitive psychology experiments examining how people learn from multimodal information sources. Additionally, *Basic Machines* has a set of homework assignments provided by the publisher that are designed to test comprehension. This homework set forms the basis of the system evaluation. The examples and evaluation presented here cover Chapter one of *Basic Machines*: Levers and the associated homework questions.

5.2.1 TEXT

The contents of a given chapter can be broken down into text content and diagram content. The text is further divided into basic informational text (which is itself divided into paragraphs), and worked example problems. Example problems typically involve a short paragraph of background information followed by a formula and substitutions of values from the background into the formula. Figure 30 shows an example of a worked solution from Chapter 1 of *Basic Machines*.

Now let's take another problem and see how it works out. Suppose you want to pry up the lid of a paint can (fig. 1-8) with a 6-inch file scraper, and you know that the average force holding the lid is 50 pounds. If the distance from the edge of the paint can to the edge of the cover is 1 inch, what force will you have to apply on the end of the file scraper?

According to the formula,

$$\frac{L}{l} = \frac{R}{E},$$

here,

$$L = 5 \text{ inches}$$

$$l = 1 \text{ inch}$$

$$R = 50 \text{ pounds, and}$$

E is unknown.

Then, substituting the numbers in their proper places, we have

$$\frac{5}{1} = \frac{50}{E}$$

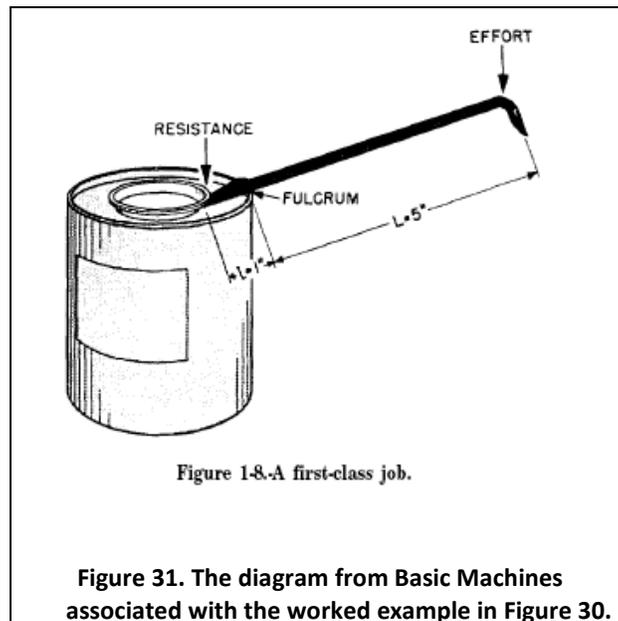
and

$$E = \frac{50 \times 1}{5} = 10 \text{ pounds}$$

You will need to apply a force of only 10 pounds.

Figure 30. A worked example problem from the text

In addition to information about levers, Chapter 1 of *Basic Machines* contains a general introduction to the whole book and a chapter summary. These two sections are ignored in this evaluation; plans to address the summary are included in future work. The remaining text includes 28 paragraphs with an average length of 3.89 sentences each. Chapter 1 also contains 9 worked examples. There are a total of 15 diagrams, 7 of which are associated with worked examples in one form or another. Figure 31 shows the diagram associated with the worked example in Figure 30. All of the diagrams from Chapter 1 can be found in Appendix J.



5.2.2 QUESTIONS

The first 29 questions of *Basic Machines* Assignment 1 address the content of Chapter 1. All of the questions are either TRUE/FALSE or multiple choice and can be found in Appendix K.

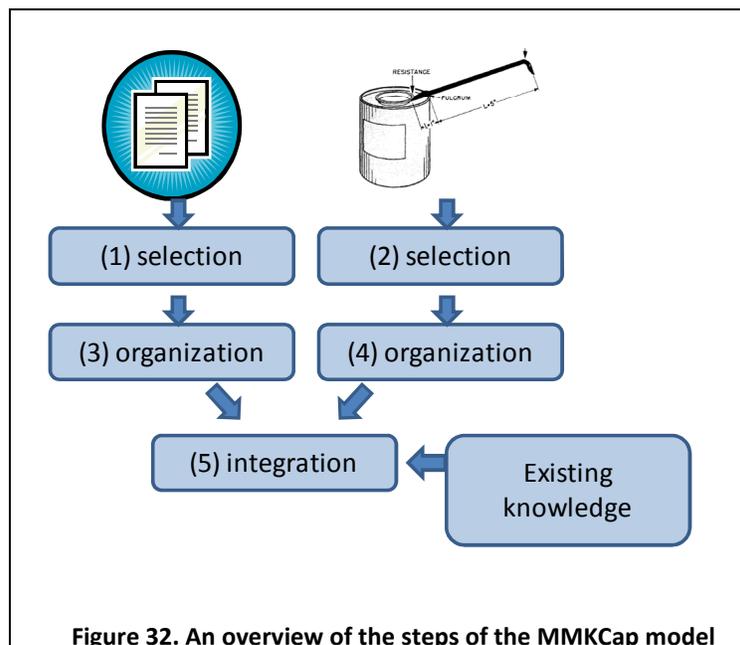
5.3 THE MMKCAP MODEL

5.3.1 OVERVIEW

This section steps through the multimodal knowledge capture (MMKCap) model, using examples from *Basic Machines* to illustrate the different steps. Each step from Mayer's multimedia learning theory, i.e.:

- 1) Selecting relevant words for processing in verbal working memory
- 2) Selecting relevant images for processing in visual working memory
- 3) Organizing selected words into verbal mental model
- 4) Organizing selected images into visual mental model
- 5) Integrating verbal and visual representations along with existing knowledge.

has a corresponding step in the MMKCap model. Figure 32 shows an overview of this process.



5.3.2 SELECTING RELEVANT WORDS AND IMAGES (STEPS 1 AND 2)

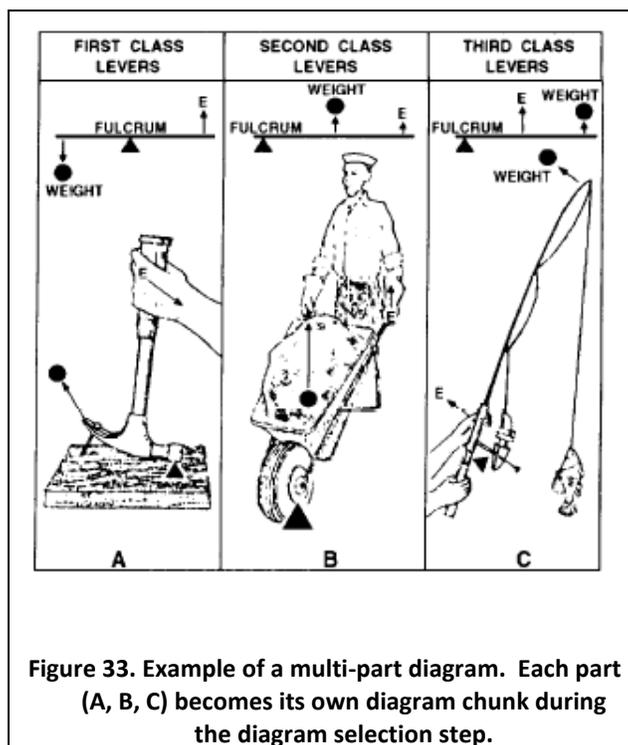
In Mayer's theory, steps one and two involve the learner deciding which portions of the text and diagrams to attend to in working memory. Selection is necessary due to limits on working memory capacity. Selection serves a similar purpose in MMKCap, to organize the information into discrete, coherent chunks. Currently, this step is done manually. A text chunk is determined by the structure of the source material according to the following rules:

- (1) Each paragraph in the original book is a separate chunk.
- (2) A paragraph may be further sub-divided if there is a reference to a figure that pertains to part of the paragraph and not to another.

Rule (2) is necessary for paragraphs where only a portion of the text refers to the diagram. In keeping with Mayer's model, only the relevant portions of the paragraph will be integrated with the diagram in the multimodal integration step. The other part of the paragraph will be separately processed as a text-only chunk.

Diagram selection is also done manually following a similar set of rules:

- (1) Each diagram in the text is an individual diagram chunk
- (2) Complex, multi-part diagrams may be separated into individual chunks (one per part) if they are referred to separately in the text. An example of a multi-part diagram is shown in Figure 33. Each of the three sections becomes its own diagram chunk.



5.3.3 ORGANIZING SELECTED WORDS (STEP 3)

Step 3 of Mayer's model involves translating the selected chunk of text into an internal representation. In MMKCap, step 3 also creates a reasoning-ready representation from the original text. Representation in this case is done using EA NLU (see Chapter 3 for a description). The input to EA during this step is QRG-CE, a simplified, natural language representation of the original text. The output is a case (called a *discourse case*) of predicate calculus facts representing the semantic content of the text.

While translating from the original text to QRG-CE does involve human intervention, it is far easier than hand-converting directly to predicate calculus. This simplification step was included to get around potential parsing difficulties and to allow this dissertation to focus on the multimodal aspects of

knowledge capture. A strict set of guidelines are used when doing the translation to QRG-CE to ensure that the simplified text adheres as closely as possible to the original:

- 1) Sentences that do not contribute topical information can be deleted: e.g. “[Machines] have taken much of the backache and drudgery out of the sailor’s lift.” or “Machines are your friends.” Future versions should distinguish between useful and non-content sentences automatically, but for now this is done by hand.
- 2) Long sentences or sentences containing conjunctions are broken into shorter, easier to parse sentences. (see Figure 34 for an example of original text and its simplified counterpart)
- 3) The mathematical steps in worked examples in the text (involving equations and numerical substitutions) are hand-represented in predicate calculus to make the steps in the problem solving process clear, and available for later use. This is another step that will be automated in the future.
- 4) Summary information at the end of the chapter is excluded for now. This information is redundant (by definition). Later a method will be developed to use summary information as a first pass check of knowledge capture
- 5) Vocabulary is standardized to create a more cohesive understanding of the information. For example, in chapter 1 of *Basic Machines*, the words “weight” and “resistance” are used interchangeably to refer to the load on a lever. To make the text clearer, “weight” was always substituted for “resistance” in the text. In future versions of MMKCap, there will be methods for automatically recognizing and handling this type of vocabulary resolution.

Figure 34 shows an example of a text chunk and diagram chunk from the original text along with the QRG-CE representation of the text content. For processing, the original paragraph has been further

subdivided into two chunks: chunk 1 contains the text that doesn't refer to the diagram and chunk 2 contains the text that does directly require the diagram. Figure 35 shows the discourse case output from EA after processing chunk 2. Processing was done using the user interface for EA where parsing and interpretation ambiguities are manually disambiguated (see Figure 36 for an example of this interface). Then the final discourse case is stored using a preexisting function. If the original text referenced a diagram, a fact is added to the discourse case indicating which diagram should be integrated with the text (e.g. the `sketchForDiscourse` fact in Figure 35).

You will find that all levers have three basic parts: the fulcrum (F), a force or effort (E), and a resistance (R). Look at the lever in figure 1-1. You see the pivotal point (fulcrum) (F); the effort (E), which is applied at a distance (A) from the fulcrum; and a resistance (R), which acts at a distance (a) from the fulcrum. Distances A and a are the arms of the lever.

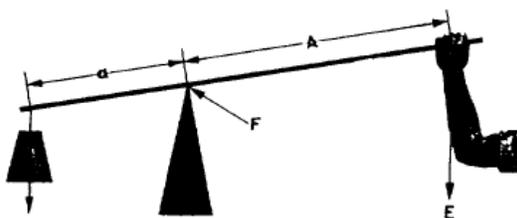


Figure 1-1.-A simple lever.

CHUNK 1

A lever has three basic parts.
 A fulcrum is a basic part of a lever.
 A force is a basic part of a lever.
 A weight is a basic part of a lever.

CHUNK 2

F is the Fulcrum
 E is the force
 R is the weight
 A2 is the distance between the weight and the fulcrum.
 A1 is the distance between the force and the fulcrum.
 A1 is an arm of the lever.
 A2 is an arm of the lever.

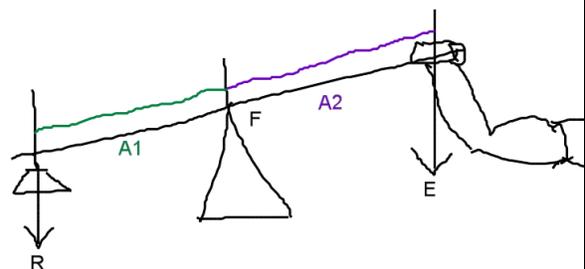


Figure 34. An example of a paragraph from the text and its associated diagram. In the second column, the text has been translated to QRG-CE and the diagram has been sketched in CogSketch.

```

(isa lever6354 Lever)
(isa a2 LeverArm)
(possessiveRelation lever6354 a2)
(isa f Fulcrum)
(isa e ForceVector)
(isa r Weight)
(isa a2 Distance)
(between r f a2)
(isa a1 Distance)
(between e f a1)
(isa a1 LeverArm)
(isa lever6231 Lever)
(possessiveRelation lever6231 a1)
(possessiveRelation lever6231 a2)
(sketchForDiscourse " Figure1-1.sk" (DrsCaseFn DRS-3446218074-8197))

```

Figure 35. The discourse case for chunk 2 of text in Figure 34

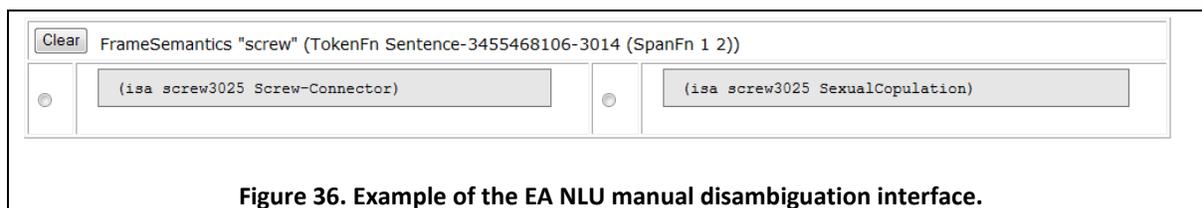


Figure 36. Example of the EA NLU manual disambiguation interface.

During processing, EA attempts to tie entities in the text to collections in the knowledge base and takes advantage of several NL-specific types of facts to aid in parsing and interpretation. For some of the concepts in *Basic Machines*, the NL-specific facts or the underlying collections were not in the knowledge base, so additional knowledge was added to facilitate interpretation. All of the added knowledge can be found in Appendix L. The majority of the added knowledge was necessary for one of the following reasons:

- A collection was needed for a concept, or sense of a concept, that did not previously exist in the knowledge base.

- Multi-word string facts were added to alert EA that a given string should be processed as one entity (e.g. “mechanical advantage” and “resistance arm”)
- Facts providing part-of-speech information
- Denotation information tying an English word to a concept in the knowledge base

Some concepts needed to be completely created from scratch. For example, the word “Seesaw” was not recognized in the lexicon and a corresponding collection did not exist in the knowledge base, so both types of information needed to be added. Other concepts may have been in the lexicon, but did not have a collection in the knowledge base corresponding to the correct sense of the word. For example, the word “Wedge” is in the lexicon, but the concepts in the knowledge base refer to `Wedge-TheSandwich` and `Wedge-TheGolfClub`, so a new collection had to be added for `Wedge-TheTool`. Currently all new concepts and lexical information must be added by hand, however the future work section discusses some ideas for automating this process in the future.

After each discourse case is created, some additional facts are created to provide book-keeping information. A `firstDRSForChapter` fact is added to the first discourse case for each chapter. Each subsequent discourse case contains a `previousDRSInChapter` fact to preserve the order of the discourse cases during the integration step. These facts also tie each discourse to the chapter in which it occurs. All of the discourse cases for each chapter are stored together in one `.meld` file for replicability as well as being stored in the knowledge base. The original sentences are discarded.

5.3.4 ORGANIZING SELECTED PICTURES (STEP 4)

Like the previous step, step four in multimedia learning involves turning the selected modality into an internal representation. The `CogSketch` sketch understanding system is used to create representations from the textbook diagrams. First, each diagram is sketched using `CogSketch`. This simplifies the image interpretation problem, allowing this work to focus on reasoning with the

information in the diagram. Like the translation of the text into QRG-CE, a series of rules governs the translation of diagrams into sketches to ensure that the original diagram is maintained as faithfully as possible and that only information present in the original diagram is created in the sketched version:

- 1) Objects in the sketch are drawn to preserve existing spatial relationships
- 2) Objects in the sketch are only labeled in the sketch if they are labeled in the source diagram
- 3) Objects labeled with conceptual information in the source diagram (i.e. Fulcrum) are given a conceptual label in CogSketch
- 4) Objects that only have identifying labels in the original source (i.e. "A") are labeled using the glyph name in CogSketch. Note that names in CogSketch are case insensitive, so labels in the text that rely on case sensitivity must be changed to be distinguishable in CogSketch
- 5) If an object is labeled with a numerical value in the source material then it is drawn as an annotation glyph in CogSketch with that numerical value.

One departure from this set of guidelines is that items are also labeled if they are clearly meant to be recognized by their shape (e.g. triangles used to denote fulcrums). This is done because currently domain-specific recognition capabilities are not built into the CogSketch system.

After each sketch has been created, it is exported as a case of facts representing the objects in the sketch and the spatial relationships between them. As part of storing each sketch, a function is called which recomputes all of the qualitative spatial relationships calculated by CogSketch. In addition to the regularly computed relationships, we also include the annotation glyphs, which is a departure from regular CogSketch operation. Some of the spatial relationships calculated are redundant and can overwhelm the SME matches performed during the evaluation, so they are filtered out. At this point,

bookkeeping information is also filtered out of the cases. All filtering is done using the filter functions found in Appendix M. The remaining facts are stored as a case in the knowledge base containing information about the objects in the sketch and the spatial relationships between them. Figure 37 shows several types of facts included in the case for the sketch in Figure 34.

```
(enclosesHorizontally (GlyphFn Object-4 Layer-2)
                      (GlyphFn Object-147 Layer-2))
(rcc8-EC (GlyphFn Object-4 Layer-2)
         (GlyphFn Object-141 Layer-2))
(visualQuantityQuantitativeMeasurement((ConceptKnownAsFn "A1")
                                       (GlyphFn Object-4 Layer-2)) A1)
(above Object-145 Object-4)
```

Figure 37. Several facts from the diagram case created for the diagram in Figure 34

5.3.5 INTEGRATION (STEP 5)

Integration in our MMKCap model is done using the Structure-Mapping Engine (SME, see Chapter 3 for a description) and a set of automatic preprocessing routines that prepare the cases for integration. The system iterates through the discourse cases in order, checking for `sketchForDiscourse` facts which indicate that there is a sketch associated with the given discourse. If a discourse case has no `sketchForDiscourse` fact, the discourse case itself is considered the final outcome of the multimodal knowledge capture for that portion of the text. If there is a diagram associated with a discourse, the corresponding sketch case is retrieved and the two cases are integrated.

The first step of integration is a preprocessing routine that checks for correspondences between the text and the diagram. Required correspondences are created between objects in the sketch case and the discourse case to restrict the SME mapping (the corresponding objects must align in the mapping) as follows:

- 1) An entity in the discourse case is identified by the same string as an entity in the sketch case. For example, in Figure 34, the text contains the following: “You see the pivotal

point (fulcrum) (F)” and the corresponding diagram has an entity (the fulcrum) labeled “F”. In this case, a required correspondence is created between “f” in the discourse case and the glyph in the sketch case that has the name “F”.

- 2) There is a spatial preposition in the text, such as “The fulcrum is *between* the weight and the force”. If there are objects in the sketch with the same types as the role fillers in the spatial relationship (here: fulcrum, weight and force) then a correspondence is created between the object in the text and the glyph in the sketch with the same type.
- 3) A running list of *chapter correspondences* is created based on the running results from #1 that contains label/type pairs (e.g. “F”/Fulcrum). This list was added to accommodate the fact that often the same label will be used in diagrams throughout the chapter (e.g. “F” to indicate the fulcrum). Entries to this list are created during the check for label-based correspondences, the label, along with the Collection associated with it is stored in the list. Later, if the label appears in a sketch, the glyph with that label will be placed in a required correspondence with any item in the discourse case that is of the type (e.g. a correspondence would be created between a glyph labeled “F” and an entity in the discourse case that was of type “Fulcrum”)

There is one additional preprocessing step that does not involve the creation of required correspondences. If the discourse case contains both (1) an `implies-DrsDrs` fact where the antecedent is of the form `(isa ?entity ?Collection)` and (2) a reference to a diagram, then that discourse is considered to contain information that is core to the concept `?Collection`. In this case, the diagram is also considered to be a canonical example of `?Collection` and a fact is added to the discourse case of the form `(sketchForConcept ?diagram ?Collection)`. Figure 38 below

shows an example of a discourse that fits this pattern with the important facts colored red. In this case, the diagram (Figure1-2) is tagged as being a good example of the concept first-class lever.

```
(sketchForDiscourse "Figure1-2A.sk" (ExplicitCaseFn DRS-3447867971-7696))
(discourseForChapter (ExplicitCaseFn DRS-3447867971-7696) "BMChapter1")
(previousDRSInChapter DRS-3447867971-7696 DRS-3447867185-7342 "BMChapter1")

(drsImplies (DrsCaseFn DRS-3447867971-7697) (DrsCaseFn DRS-3447867972-7698))

(in-microtheory DRS-3447867971-7697 :exclude-globals t)
(genlMt DRS-3447867971-7697 Chapter1RawText)
(isa first-class-lever7447 Lever-FirstClass)

(in-microtheory DRS-3447867972-7698 :exclude-globals t)
(genlMt DRS-3447867972-7698 Chapter1RawText)
(between force7539 weight7589 fulcrum7487)
(isa force7539 ForceVector)
(isa fulcrum7487 Fulcrum)
(isa weight7589 Weight)
```

Figure 38. An example discourse case showing how sketchForDiscourse facts are created.

After the required correspondences are created (there may not be any for a given discourse-diagram pair), an SME match is run with the diagram case as the target and the discourse case as the base. An integrated case is created which contains the contents of the diagram case plus the candidate inferences from the SME match. The candidate inferences represent the information from the text that is connected to the diagram. Figure 39 below shows the candidate inferences from the match between the discourse and diagram cases created by the example in Figure 34. Integrated cases are stored in the knowledge base and are also dumped to a text file for archiving. Worked examples from the text are integrated using the same algorithm as the regular text.

```

(isa Object-416 Weight)
(isa Object-412 Fulcrum)
(isa Object-434 LeverArm)
(isa Object-422 ForceVector)
(isa Object-430 Distance)
(possessiveRelation (AnalogySkolemFn lever6231) Object-434)
(possessiveRelation (AnalogySkolemFn lever6231) Object-430)
(between Object-422 Object-412 Object-430)
(between Object-416 Object-412 Object-434)
(possessiveRelation (AnalogySkolemFn lever6354) Object-434)
(isa Object-430 LeverArm)
(isa Object-434 Distance)

```

Figure 39. Candidate inferences from the integration of the discourse and diagram cases from the example in Figure 34

5.4 EVALUATION

The MMKCap system was evaluated using a publisher-provided set of homework problems. To facilitate the evaluation, a test-harness was built to query for the information in each question and to test the correctness of the system's response. To eliminate errors in question comprehension or problem solving, each question and the associated multiple choice answers were hand-translated into predicate calculus and stored in the knowledge base. Each question was also tagged with a fact tying it to the appropriate chapter. Questions that referred to a diagram had an added fact that referred to the diagram and the diagram itself was sketched using CogSketch and stored as a sketch case using the procedure used for diagrams in the original text. Each question in the homework set was categorized into one of six question types. The test harness retrieves all of the questions for a chapter and then applies a different problem-solving technique depending on the question type. An additional fact is created for each question indicating which of the multiple choice options is correct.

5.4.1 QUESTION TYPES AND ANSWER STRATEGIES

Each question in the test set was tagged with a fact indicating which of six question types it represents. These facts aid the problem solving system in picking the correct solving strategy for each question. The six types are:

- 1) True/False
- 2) Simple Query: A factual multiple-choice question
- 3) Diagram-Concept: Pick the diagram that matches the concept (e.g. first-class lever)
- 4) Diagram-Measurement: Read off a measurement from a diagram
- 5) Algebraic: Use the formulas in the chapter to work out a solution to the problem
- 6) Algebraic + Diagram: Same as #5, but with a diagram that provides some of the necessary information for the question

This section describes each question type in detail and discusses the technique used by the solver to answer each type of question. The questions are multiple-choice, but the problem solving system was not allowed to guess if it did not find an answer.

The first type of question is a typical TRUE/FALSE question. Scenario information may be supplied and then the test-taker must determine whether a given statement is true or false. Figure 40 shows an example of a TRUE/FALSE question from the test set and its translation into predicate calculus. The test harness simply queries for the information in the question and selects true if it is found in the knowledge base and false otherwise. The selected answer is compared to the correct answer to determine whether the question was answered correctly.

1-2. When a chain hoist is used to multiply the force being exerted on a load, the chain is pulled at a faster rate than the load travels.

1. True
2. False

```
(querySentenceOfQuery BasicMachines1-2
  (implies
    (isa ?hoist Pulley)
    (and
      (isa ?load Weight)
      (isa ?force ForceVector)
      (isa ?travel MovementProcess)
      (isa ?travel ComparisonEvent)
      (comparee ?travel ?force)
      (comparer ?travel ?load)
      (comparativeRelation ?travel
        (HighAmountOfFn Speed))))))
```

Figure 40. Example of a True/False question

Figure 41 shows an example of a simple query type of question. Simple query questions are straightforward factual questions based on the material in the chapter. Like TRUE/FALSE questions, simple query questions are solved by simply querying for the requested information. The results from the query are compared to the list of possible answers. If an option matches, it is chosen as the answer. If none of the multiple choice answers match, the system does not answer. In some of the questions, the multiple choice answers provide several potentially correct answers. For example one of the questions provides the following multiple choice options: “first-class lever”, “second-class lever”, “first- or second-class lever”. In cases like this, the option that has the highest overlap with the answer returned by the query is chosen as the answer for the question.

1-5. Which of the following simple machines works on the same principle as the inclined plane?

1. Screw
2. Gear
3. Wheel and axle
4. Block and tackle

```
(termToSolveFor
  (querySentenceOfQuery
    BasicMachines1-5
    (and (refersToTypeOf ?tool InclinedPlane)
      (isa ?tool ?collection)))
  BasicMachines1-5
  ?collection)
```

Figure 41. Example of a simple query question

For the most part, simple query and TRUE/FALSE questions rely on knowledge captured from the text to construct an answer. In most cases, they do not take advantage of knowledge resulting from the multimodal integration portion of the MMKCap model. However, they do demonstrate the effectiveness of EA NLU in capturing knowledge from the text portion of the materials. Also, these questions show that the multimodal integration does not interfere with the knowledge capture from the text.

The next two types of questions, diagram-concept and diagram-measurement, would be impossible to answer, in many cases, without multimodal information from the text. In the first case diagram-concept, the question gives a set of pictures and asks the test taker to select the picture(s) that correspond to a given concept. In the example in Figure 42, the question is which picture(s) illustrate a second class lever. To solve this problem, the system searches through the text for a known example of a second-class lever. Priority is given to retrieved instances of integrated cases that contain a `sketchForConcept` fact (created during the integration process). If no `sketchForConcept` fact is found, then the system will pick any integrated case that contains a diagram and has facts about second-class levers. In the example in Figure 42, a `sketchForConcept` fact is found and the integrated case containing information about the sketch in Figure 43 is retrieved.

Figure 1A.

QUESTIONS 1-7 THROUGH 1-9 RELATE TO THE DRAWINGS IN FIGURE 1A.

1-8. Which part illustrates a Second-class lever?

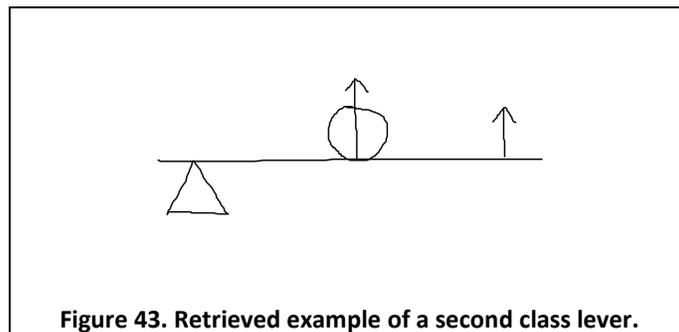
1. D
2. C
3. B
4. A

```

(termToSolveFor
 (matchConceptToPicture Lever-SecondClass
 (TheSet
 (sketchForQuery "Question1-7A.sk"
 BasicMachines1-8)
 (sketchForQuery "Question1-7B.sk"
 BasicMachines1-8)
 (sketchForQuery "Question1-7C.sk"
 BasicMachines1-8)
 (sketchForQuery "Question1-7D.sk"
 BasicMachines1-8)))
 BasicMachines1-8
 ?sketch)
    
```

Figure 42. Example of a diagram-concept question

The sketch from the integrated case retrieved is then compared to each of the sketches in the problem description using SME. In this example, the sketch in Figure 43 would be compared to each of the sketches (A-D) in the problem. The sketch that matches with the highest structural evaluation score is chosen as the answer for the problem.



In diagram-measurement questions, like the one shown in Figure 44, the task is to use the diagram to find the numerical (or symbolic) measurement for a given component. In the example pictured, the length of the resistance arm is the quantity that is asked for. To solve this type of question, the system searches for an integrated case that contains a reference to an object of the correct type (in this example resistance arm). This reference can either be in the form of an isa: (isa x ResistanceArm-LeverArm) or a glyph name in a sketch. The second option is a side-effect of the interface for creating annotations in CogSketch. When an annotation is created, only one collection can be used to specify the type and that must be one of the designated annotation types (e.g. LengthIndicator), so if an object also needs to be labeled as something else (e.g. a Resistance Arm) this must be done using the name slot for the glyph.

Continuing the running example, once all of the integrated cases containing a resistance arm are retrieved, the best match between the retrieved cases and the problem sketch is found (again using the SME structural evaluation score to determine best match), the best matched sketch is shown on the right in Figure 45. A new mapping is done between the best matched retrieved case and the problem sketch. The mapping is examined for a correspondence between the glyph labeled as the resistance arm in the retrieved case and a glyph in the problem sketch. In this case a correspondence is found between the resistance arm and Obejct-121 in the problem sketch, so the numerical value associated with that object is returned as the answer to the question. In this case, that value is 1ft. which is the correct answer.

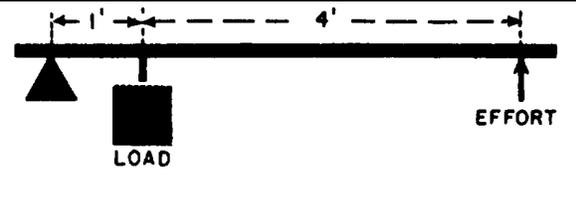
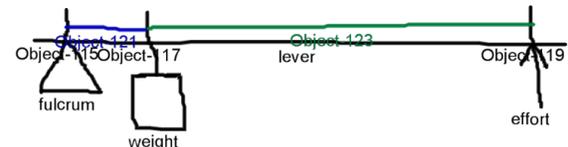


Figure 1C

IN ANSWERING QUESTIONS 1-12 THROUGH 1-14, SELECT THE CORRECT ARM MEASUREMENTS FROM FIGURES 1B AND 1C.

1-14. Resistance arm in figure 1C

1. 1 ft
2. 3 ft
3. 4 ft
4. 5 ft



```

(sketchForQuery "Questions1C.sk"
  BasicMachines1-14)
(determineMeasurementFromSketch
  ResistanceArm-LeverArm
  BasicMachines1-14)

```

Figure 44. Example of a Diagram-Measurement Question. The diagram needed for the question is sketched using CogSketch and a fact is created indicating which measurement is needed.

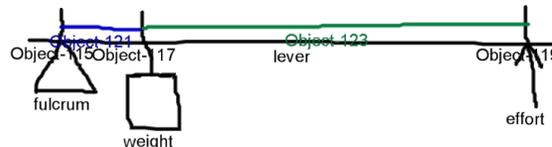




Figure 45. Example of the sketch from the problem on the left and the retrieved best match on the right. An SME mapping is done between the two cases and the glyph in the sketch on the left that is in correspondence with the ResistanceArm glyph (indicated in purple) in the sketch on the right is determined. In this case, the corresponding glyph is Object-121 (blue glyph), so its numerical value 1ft is returned as the answer.

The next two types of questions both involve working out algebraic solutions to problems using the formulas from the chapter. In algebraic questions, background information needed for the question is presented in text. In algebraic+diagram questions, the background information needed to solve the problem is presented in a combination of text and an accompanying diagram. In both types of problems, the algorithm for finding a solution is:

- 1) Use MAC/FAC to retrieve the best match from among the worked examples in the chapter text (a worked example may also have contained text only or a text-diagram pair)
- 2) Use SME to do a mapping between the retrieved example (base) and problem (target)
- 3) Use candidate inferences from the mapping to determine the equation and value substitutions to use to solve the problem
- 4) Make the value substitutions into the equation
- 5) Use a pre-existing algebraic problem solver to solve the equation for the desired value

<p>Questions 1-17 and 1-18 are related to a 300-pound load of firebrick stacked on a wheelbarrow. Assume that the weight of the firebrick is centered at a point and the barrow axle is 1 1/2 feet forward of the point.</p> <p>1-17. If a Seaman grips the barrow handles at a distance of three feet from the point, how many total pounds will the Seaman have to lift to move the barrow?</p> <ol style="list-style-type: none"> 1. 65 lb 2. 100 lb 3. 150 lb 4. 300 lb 	<pre>(isa Wheelbarrow1-17 Wheelbarrow) (isa BarrowAxle1-17 Axle) (isa BarrowAxle1-17 Fulcrum) (isa LoadOfBricks1-17 Weight) (valueOf LoadOfBricks1-17 (Pound- UnitOfForce 300)) (isa distance1-17b Distance) (distanceBetween LoadOfBricks1-17 BarrowAxle1-17 distance1-17b) (valueOf distance1-17b (Foot-UnitOfMeasure 1.5)) (isa Seaman1-17 ForceVector) (isa distance1-17a Distance) (distanceBetween Seaman1-17 BarrowAxle1-17 distance1-17a) (valueOf distance1-17a (Foot-UnitOfMeasure 3)) (querySentenceOfQuery BasicMachines1-17 (valueOf Seaman1-17 ForceNeeded1-17))</pre>
---	---

Figure 46. Example of an algebraic question

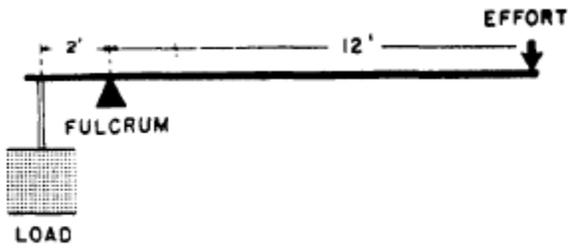
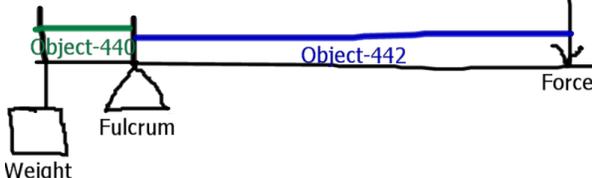


Figure 1J



1-26. The mechanical advantage of the lever pictured in figure 1J is

1. five
2. six
3. seven
4. one-sixth

```

(sketchForQuery "Question1J.sk"
  BasicMachines1-26)

(isa lever1-26 Lever)

(querySentenceOfQuery
  BasicMachines1-26
  (valueOf (mechanicalAdvantageOf
    lever1-26) MA1-26))
  
```

Figure 47. Example of an Algebraic w/Diagram Question

In all of the question types, the answer returned by the system is compared to the known correct answer to determine whether the question was correctly solved.

5.5 RESULTS

Table 11 shows a break-down of the number of correct questions by query type for the evaluation of Chapter 1 of *Basic Machines*.

Table 11. Summary of evaluation results

Question Type	Total Number	Number Correct	p-value
TRUE/FALSE	2	2	0.25
Simple Query	9	9	$< 10^{-5}$
Diagram-Concept	3	2	0.16
Diagram-Measurement	4	2	0.09
Algebraic	6	4	N/A
Algebraic + Diagram	5	1	N/A
TOTAL	29	20	$< 10^{-6*}$

* total p-value excludes algebraic questions

A close examination of the failures provides additional insights and suggests improvements. The diagram/concept matching question that fails is the example in Figure 48. Figure 48 also shows the diagram that is retrieved as the example of a third class lever from the text. It is compared (via SME) to the sketched versions of the levers in the problem. The system should match to option A (shown circled with a solid line), which is an example of a third class lever, but instead matches to option C (shown circled with a dashed line), which is an example of a first class lever. This mistake occurs because our system currently cannot recognize that the lever in option A needs to be flipped over the horizontal axis, so that the fulcrum is underneath the lever in order for the comparison to work correctly. As the pictures are currently drawn, the matching system is overwhelmed by the number of spatial relationships in common between the retrieved sketch and the lever in option C (e.g. the fulcrum being below the lever and the force and weight being above) and those relations override the important one, which is the placement of the fulcrum relative to the weight and effort. This suggests verifying properties of the match, i.e., that forces are being applied in ways consistent with the learned definition. If verification fails, then rerepresentation techniques based on spatial properties, like flipping or rotating one of the sketches, could be tried.

Figure 1A.

QUESTIONS 1-7 THROUGH 1-9 RELATE TO THE DRAWINGS IN FIGURE 1A.

1-8. Which part illustrates a third-class lever?

1. A
2. B
3. C
4. D

Sketched versions of the diagrams in the question

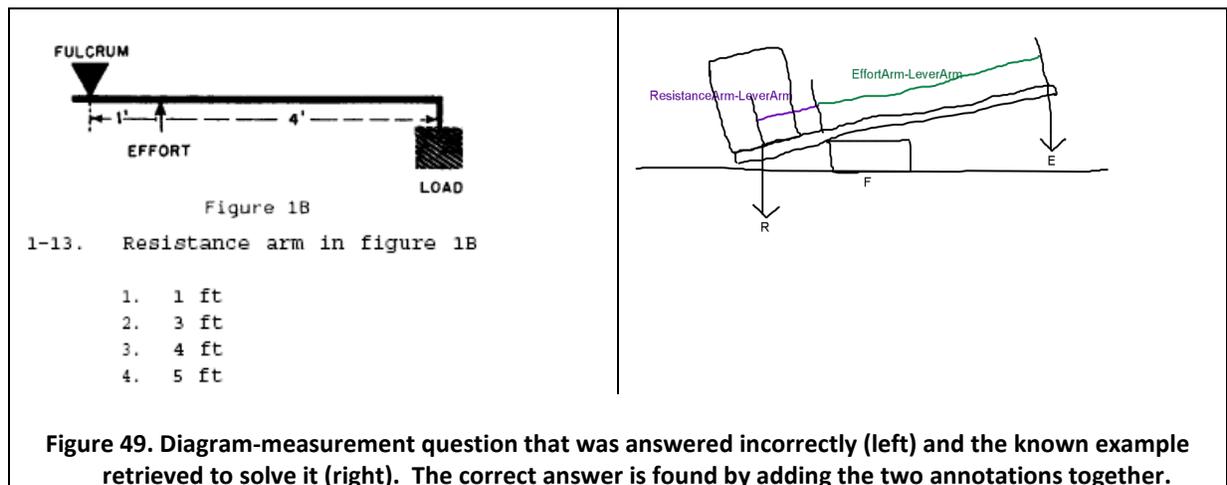
Retrieved known example of a third-class lever. Matches erroneously to option C (dashed circle) but should match option A (solid circle)

Figure 48. Example of the failed diagram-concept question

The two diagram measurement query failures illustrate another shortcoming of our problem solving strategy of using a simple, unevaluated match to produce an answer. The problem shown on the left in Figure 49 is the one of the two failed questions of this type. The Q/A system attempts to solve this type of problem by looking for integrated cases that have facts about the type of object that participates in the query (e.g. resistance arms) and that have an associated diagram. In this case it finds the case containing the diagram shown on the right in Figure 49 which has a graphical example of both a resistance arm, and an effort arm. The lever arms are annotations in both the test and retrieved cases, as they have numerical length values associated with them. The system performs an analogy between

the retrieved sketch and the test sketch and looks for the glyph in the test sketch that aligns with the resistance arm glyph in the example sketch. Unfortunately, the retrieved sketch is a different type of lever, where the resistance arm length can be read directly from the annotation, without having to add values. Our problem solver is currently unable to handle this situation and returns the wrong value.

Our current problem solver expects there to be a single annotation glyph it can match against to compute the answer, like in the retrieved sketch shown on the right. However, the correct answer is 5, which requires the addition of the two length annotations in the sketch. Here what is required is to realize that the length is the sum of the two glyphs that are in the diagram. Again, this is more a failure of our problem solver than of the knowledge capture process.



The majority of the algebraic+diagram questions fail for the same reason as the diagram-measurement queries. When SME performs a mapping, annotations get mapped to each other, even though many times the value needed in the problem actually comes from adding two annotations in the problem sketch as in Figure 50 below where the length of the effort arm is 9 inches ($8.5 + .5$). I am currently searching for a general solution to this problem, which arises in other problem solving tasks.

Procedural knowledge related to problem solving is much harder to extract from the text as it is often not explicitly stated, but must be extracted from the examples given.

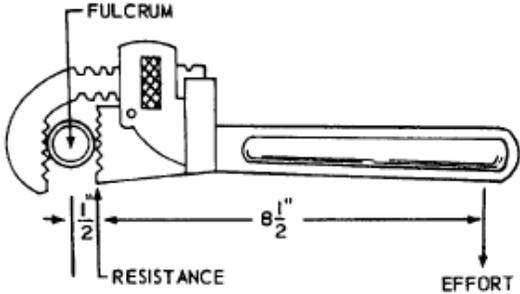


Figure 10

1-16. With the aid of the pipe wrench shown in figure 1D, how many pounds of effort will you need to exert to overcome a resistance of 900 pounds?

1. 25 lb
2. 50 lb
3. 75 lb
4. 100 lb

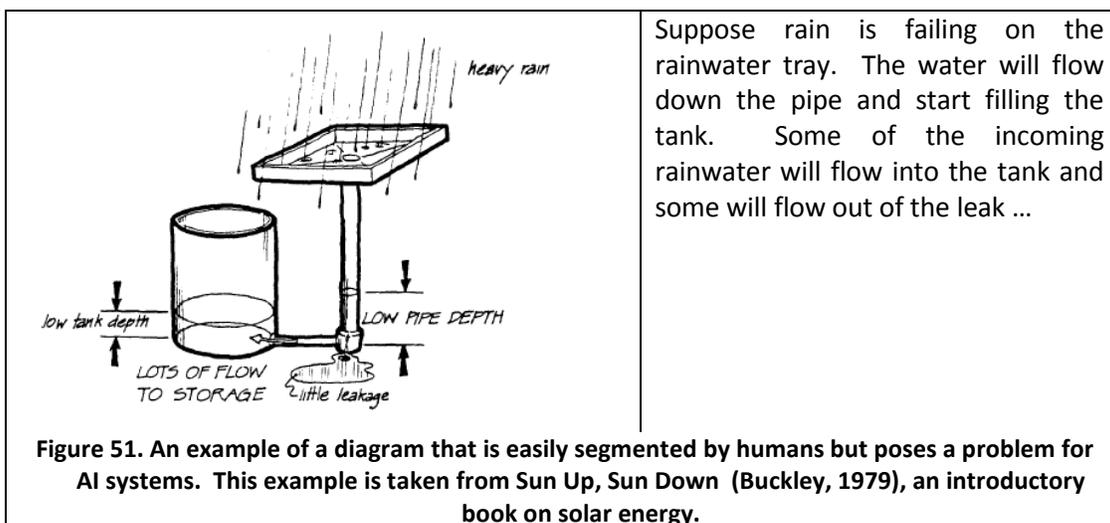
Figure 50. Example of an algebra+diagram problem that is answered incorrectly

The MMKCap system has shown that analogy is a good tool for modeling both the integration step of multimodal knowledge capture and for solving textbook problems. There are still several shortcomings of this system, both in knowledge capture and in problem solving. One of the primary issues is that the system is quite good at capturing factual knowledge, but is not as good at capturing procedural knowledge. One of the biggest challenges for future versions of the system is to improve its ability to capture and apply knowledge that relates specifically to problem solving techniques.

5.6 DIAGRAM UNDERSTANDING

5.6.1 PROBLEM DESCRIPTION: CONCEPTUAL SEGMENTATION

Part of Mayer’s multimedia learning theory involves the learner selecting the appropriate part of a diagram to attend to. Human learners get quite good at this through repeated exposure to graphical learning materials like the example in Figure 51. The caption provides a description of a process, the filling of a tank with rainwater, while the diagram provides an illustration of the physical layout of the system (tank, pipe, etc). Parsing the qualitative information in the diagram is easy for people, but quite complicated for software. For example people can both recognize the diagram as a full system and refer to its individual components like “the water in the tank”. Part of this flexibility is due to our familiarity with diagrams and their depiction conventions. Another source of flexibility is our knowledge about how things like tanks and pipes and water work. We are able to use both types of knowledge when looking at a diagram. Being able to interpret diagrams in this way is a key component of multimodal knowledge capture.



This section describes some preliminary work on conceptual segmentation of diagrams. Conceptual segmentation is defined to be the assignment of conceptual interpretations to regions and edges within the sketch. Conceptual segmentation is not currently a part of our MMKCap system, but will most likely be a necessary component when more complicated texts and diagrams are processed.

In MMKCap the diagrams are represented by their sketched equivalents. In CogSketch, glyphs are given conceptual labels which do simplify the task of conceptual segmentation. However, conceptual labeling of ink is necessary, but not sufficient, to solving this problem. Consider the sketch in Figure 52 below showing a tank partially filled with water. We will use this example throughout this section. This sketch consists of two glyphs: one closed polygon representing the tank, and one line representing the water. A literal interpretation of this sketch would be that the polygon is the tank and the squiggly line glyph itself is the water. However, the intended interpretation is that the entire area inside the tank and underneath the line is the depicted water.

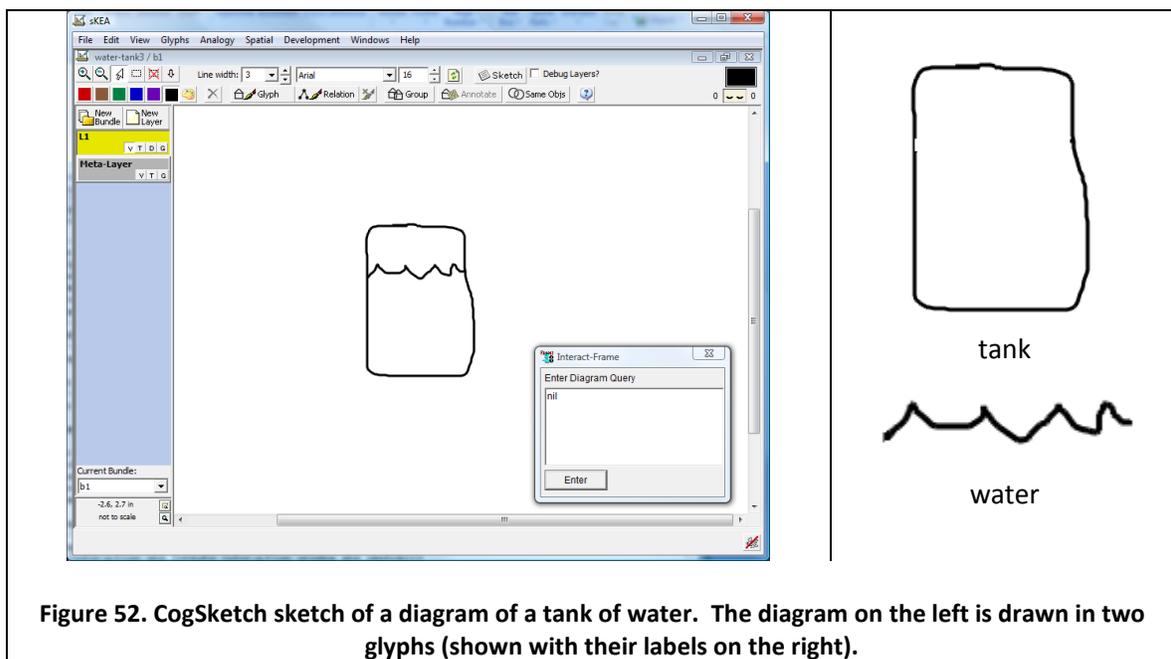
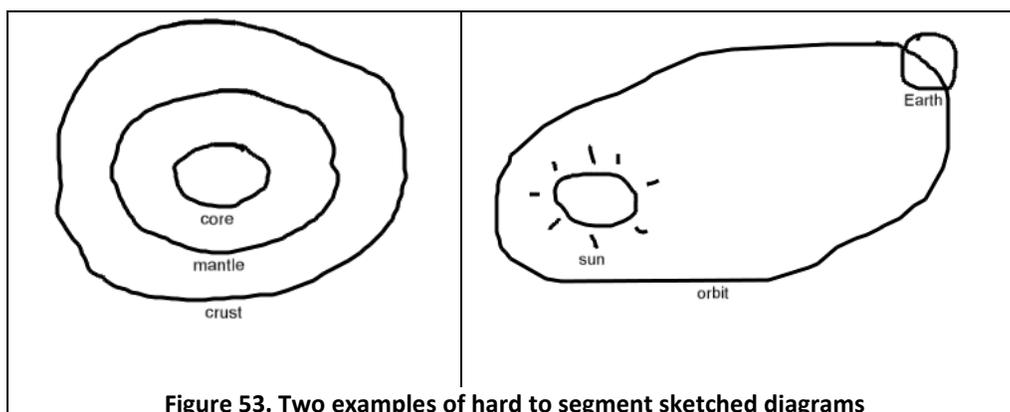


Figure 52. CogSketch sketch of a diagram of a tank of water. The diagram on the left is drawn in two glyphs (shown with their labels on the right).

One way to address this would be to require users to draw following specific conventions – for example, have them trace around the inside of all of the tank/pipe glyphs so that the water glyph was one continuous closed shape. However, while this constraint could be instituted, it only addresses this specific situation, and adding new constraints to address every new situation is untenable. Additionally, requiring users to trace the full outline of the water still leaves the situation ambiguous. The system still doesn't have a way to figure out if the user intended just the outline to represent water, or all of the space contained by the outline. For example, consider the two sketches in Figure 53. In the sketch on the left, depicting the layers of the Earth, each layer (e.g. the mantle) is represented not by the entire area inside the glyph depicting the mantle, but by the area inside that glyph but outside the outline of the core. In the sketch on the right, depicting the solar system, both the Earth and the orbit are depicted as ellipses, however the orbit is meant to be represented only by the path traced out by the ink which the planet is meant to be the whole area inside the ink for the Earth glyph. These two sketches show that both drawing conventions and statistical sketch recognition are insufficient for open-domain conceptual segmentation.



5.6.2 PRELIMINARY SYSTEM AND RESULTS

A proof-of-concept conceptual segmentation algorithm was developed to examine ways to automatically interpret diagrams. The basic algorithm is shown in Figure 54. Input to the system is a CogSketch sketch and a query term indicating the entity to find in the sketch. So, for example, in the water and tank example, the system would be given the sketch of the water and tank and the query term “water” indicating that it should find the extent of water in the sketch. The query term is matched against the glyph names in the sketch. Once the appropriate glyph is identified, we access the conceptual label(s) provided by the user. In our example, the glyph being considered is labeled with the concept `Water` from the ResearchCyc KB. Knowing what the glyph represents helps the system figure out how to interpret the diagram correctly. For example, ResearchCyc has 335 facts about water. This includes information about its role in the ResearchCyc ontology and, especially important for our purposes, some linguistic knowledge about the term.

- (1) The system is given a sketch and a query term
- (2) A check is done for a glyph corresponding to the query term
- (3) Semantic knowledge about the query term is extracted from the KB
- (4) The system decides whether the query term should be represented via an edge or region
- (5) The edges of the glyph corresponding to the query term and other involved glyphs are decomposed and rejoined to determine the correct region or edge

Figure 54. Steps in conceptual segmentation algorithm

Once the conceptual label is retrieved, conceptual information along with the ink from the sketch are used to determine the correct conceptual segmentation of the diagram. Figure 55 shows an outline of the process the system uses to determine the correct depiction for a glyph using both the conceptual label and the ink. This figure shows the algorithm as it is currently implemented; as the number and type of diagrams that interpreted is expanded, the algorithm will be further refined. In the

first step, the conceptual label is accessed and the knowledge base is queried to determine which category the entity belongs to: (1) a mass noun or entity that subclasses from the Cyc concept `TangibleStuffCompositionType` (2) an entity that subclasses from `Path-Spatial` (3) or a physical object. Backchaining rules are used to ascertain whether a concept needs a region versus a polyline to depict it. For example, a concept might contain information that, linguistically, the word referring to it is a mass noun or a count noun. Mass nouns refer to entities that can be viewed as spatially flexible pieces of stuff, such as liquids and powders, whose boundaries are highly constrained by containment relationships. The concept `Water` is linguistically a mass noun, and consequently the system infers that it requires a volume to depict it.

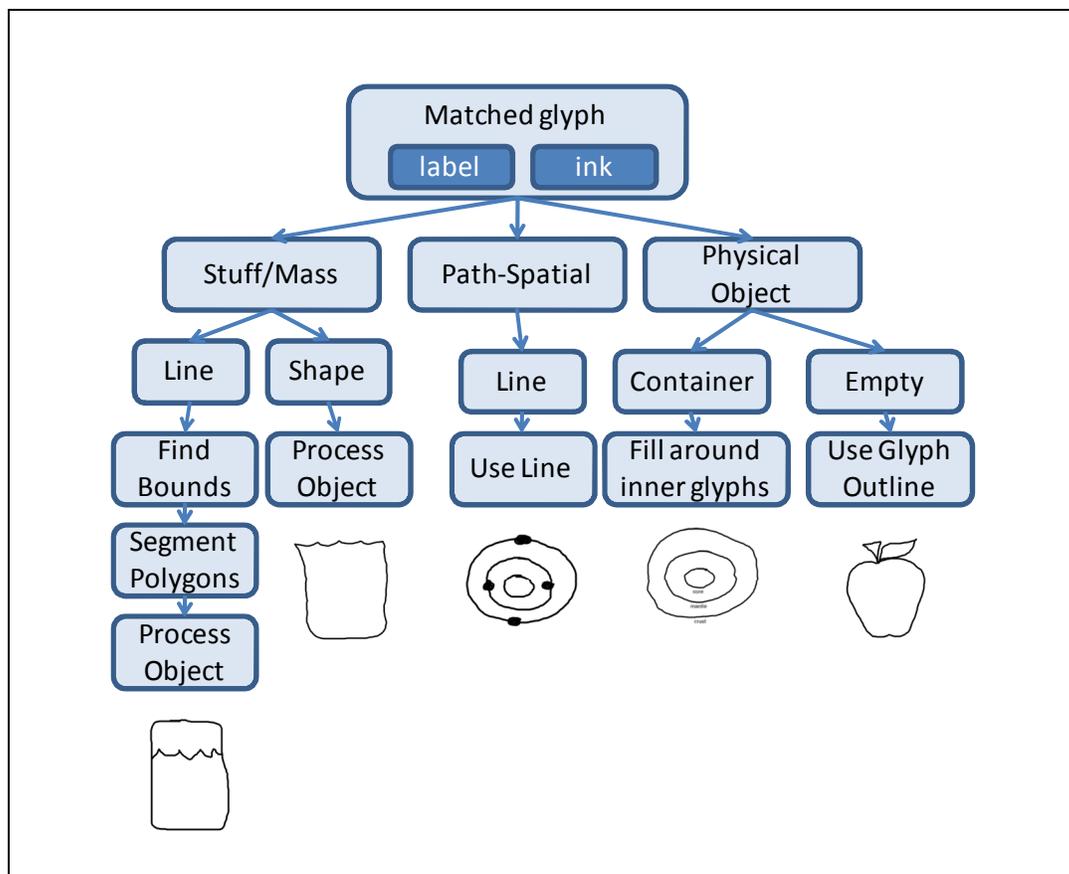


Figure 55. Decision tree of possible segmentation options for entities in diagrams.

Once the system has inferred the conceptual category for a glyph, it attempts to find or construct the appropriate geometric entity. For the water/tank example (an instance of the stuff/mass path through Figure 55) it starts by classifying the geometric properties of the ink for the glyph, determining if it is a line or a polygon. For example, the glyph representing water in Figure 52 is a line, not a polygon. Since the depiction of water requires a region, the system has more work to do. (A user could have drawn the water by tracing out a region inside the tank, in which case the system would be satisfied with the glyph itself as the geometric entity.)

The next step is to determine if there are other glyphs which can help constrain the extent of the object. In this example, the tank glyph constrains the extent. The system finds such glyphs by looking for RCC8 relationships, i.e., glyphs for which the water is either TPP or NTPP (i.e., Tangential Proper Part or Non-Tangential Proper Part). When these relationships hold between the tank glyph and the water glyph, we then do a follow-up check to see if the water intersects (within a threshold) both sides of the tank. Once both glyphs have been found (the water and the tank) the system needs to find the region representing the part of the tank where the water is found. This is accomplished by combining the ink from the two glyphs and segmenting it into *edges* and *edge cycles*.

Edges are identified by segmenting the ink at places where one line intersects another, or where there is a clear corner along a line. Edge cycles are identified by finding minimal closed cycles among the edges. In the current example, CogSketch identifies two edge cycles, one representing the area in the tank above the water and the other representing the area in the tank below the water.

For stuff/mass nouns, the system assumes the user has drawn the uppermost edge of the object, and that the object descends from there to fill the container below it. Thus, in the current example, the system looks for a cycle such that glyph for water overlaps with the top of the cycle, while the rest of the cycle is made up of points from the tank glyph. If an appropriate cycle is found, it is

identified as the region that the user is looking for, and it is then converted to a polygon and processed like a physical object.

Physical objects (the third path in Figure 55) are checked to see if they contain other glyphs (containment is one of the spatial relationships computed automatically by CogSketch). If the glyph has other objects inside of it, the algorithm as currently implemented assumes that the correct segmentation for the glyph is the space around the inner objects. This is the correct interpretation for situations like the layers of the earth, or bubbles in soda. If a physical object has no interior glyphs, the whole area of the polygon is considered the correct depiction and it is highlighted in the diagram. Figure 56 show a variety of results from running the current conceptual segmentation algorithm on different diagrams.

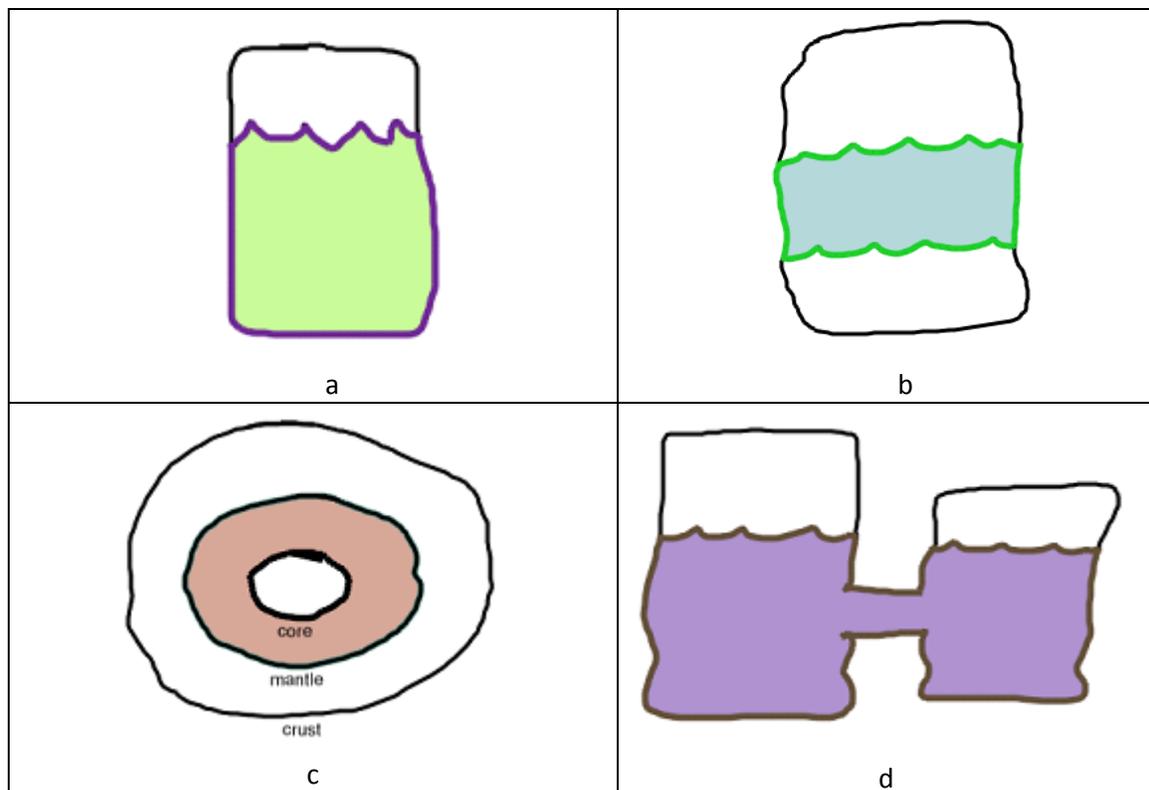


Figure 56. Results from running conceptual segmentation.

In Figure 56, part a, the results from the simple water in tank example are shown. In part b, the sketch is drawn in three glyphs, one for the tank, one wavy line labeled water, and another wavy line labeled oil. When queried for oil, the system is able to correctly identify the area between the two wavy lines as the correct extent of the oil in the sketch. Figure 56, part C shows the layers of the Earth again, when queried for “mantle”. The system correctly identifies the area between the mantle glyph and the core glyph. Currently the handling of interior glyphs is quite simple, but will have to become more sophisticated in future implementations. For example, in a sketch of a spoon in a glass of water, if the system is queried for “water” it should fill over the spoon instead of around it. Figure 56, part d shows a slightly more complicated version of the original water in tank where the water is split over two tanks connected by a pipe. The system is still able to correctly identify all of the water in the sketch.

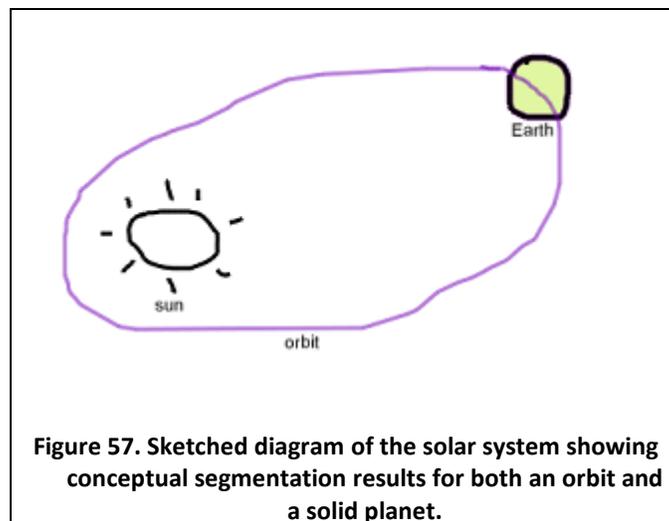
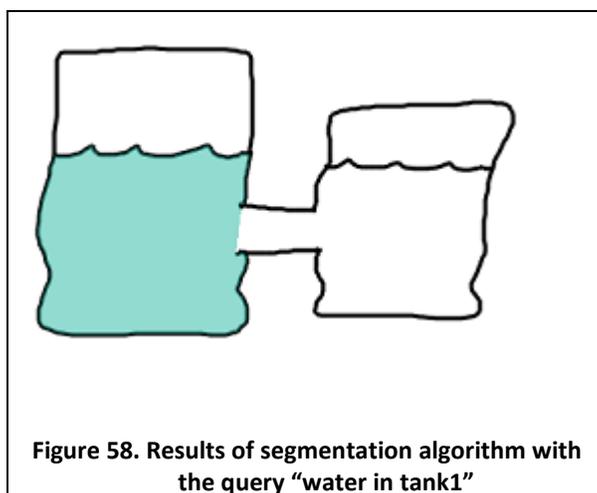


Figure 57 shows the results of querying for both “orbit” and “Earth” in the solar system sketch. The system is able to correctly differentiate between an orbit which is represented by a path and a planet which requires the entire area of the glyph. Figure 58 shows the two-tank water example, however in this case the system was queried for “water in tank1”.



This section has introduced a proof-of-concept implementation of a conceptual segmentation algorithm. Clearly, it is in very early stages and will need to be expanded in future work. However, the results so far are interesting in that they show that a combination of conceptual information and ink can be used to more intelligently segment diagrams automatically. The final example also shows how a solid understanding of spatial prepositions will play a key role in diagram segmentation.

5.7 RELATED WORK

There are three main areas of related work that are relevant to the systems in this chapter: work from the diagram understanding, multimodal knowledge capture, and learning by reading communities.

From the diagram understanding community, the division of scene elements into edges and regions in sketches was explored in the Mapsee program of Reiter and Mackworth (1989). They proposed a logical framework for depiction that formalized the mapping between images and scenes of simple maps containing roads, rivers, shores (represented as edges in the images) and water and land (represented by regions in the images). They identified a set of six visual relations ({tee, chi, bounds, closed, interior, and exterior}) and provided axioms and constraints which combined these visual

primitives and mapped them to the scene elements (roads, rivers, etc). Like Mapsee, the conceptual segmentation work is concerned with modeling how conceptual entities are depicted. However, Mapsee was designed for one domain, maps, and its axioms map visual elements directly to interpretations in that domain. By contrast, the conceptual segmentation model works through an intermediate distinction – regions versus edges – and performs reasoning over a large-scale, off-the-shelf knowledge base to identify depiction constraints. Their task was fundamentally one of image interpretation, recognizing unlabelled lines as map elements, whereas the task described here starts with conceptually labeled ink.

Alvarado and colleagues (Alvarado and Davis, 2004; Alvarado, Oltmans and Davis, 2002) describe a multi-domain sketch recognition engine. Their systems use a hierarchical shape description language where low level shape description (circles, arrows, etc) are defined once in a domain-independent fashion. Then a separate set of rules ties a given shape to a domain specific interpretation (e.g. an arrow represents a child link in a family tree diagram). This approach works well in a very tightly constrained domain with a small number of differentiated symbols (family trees, circuit diagrams, etc) however, it does not work as well in the more open-domain, unconstrained types of sketches that are encountered in multimodal knowledge capture.

Futrelle has several systems (e.g. (Futrelle, 1990)) that parse the diagrams in scientific papers. He uses a set of Generalized Equivalence Relations (GERs) like near and parallel to describe the relationships between objects in the diagram. Kara and Stahovich's SimuSketch (2004) takes a two-stage approach to recognition in sketched diagrams containing arrows. Other ink in the sketches is grouped and segmented based on clustering around the head and tail of recognized arrows. The clusters of ink are then matched against 24x24 templates for recognition. This is a unique and interesting approach to segmentation which gets around many cumbersome algorithms such as time outs or requiring single

strokes. SimuSketch is embedded in Matlab's SimuLink system which uses the recognition to simulate and solve actual engineering problems.

Approaches for diagram understanding like those above that rely on low-level shape recognition along with domain-specific rules for interpretation represent a complementary approach to the conceptual segmentation work in this chapter. A hybrid system, which combines low-level recognition for common elements (e.g., arrows) and a more generative interpretation process might be useful in many tasks. For example, in a physics system, it might be useful to automatically recognize arrows and interpret them as forces while leaving the types of objects that those forces can act on unconstrained given the wide variety of physical objects in the world.

Saund and colleagues (Saund, *et al*, 2002; Saund, 2003) have also done a lot of work on intelligently segmenting sketches. Like us, they also do not work on recognition. They focus on identifying important relationships between objects in the sketch rather than the specific objects represented. This information can be used to make intelligent decisions about which parts of a sketch a user is trying to select or edit. It also leaves open the possibility of integrating recognition later. Anderson and Armen's DiaSketch (2002) is interested in inter-diagrammatic reasoning – learning from multiple diagrams of the same information. They focus on sketching as a way of interacting with a more precisely defined diagram (such as one that was scanned in).

The MMKCap work in this chapter is related to several systems from the knowledge capture and learning by reading communities. Ferguson's JUXTA system (Ferguson and Forbus, 1995) reasoned about juxtaposition diagrams, diagrams that use comparison to illustrate physical principles, by using information from both the diagram and the caption. JUXTA relied on a pre-defined mapping from shapes to domain-specific meaning and required all of the captions to be hand-translated to qualitative physics representations. Bulko's BEATRIX system (1988) was able to solve the coreference problem for

physics problems that contained both text and a diagram. It relied on a blackboard architecture to align objects in the diagram with their references in the text. BEATRIX relied on hand-coded knowledge sources to identify potential objects in the diagrams in its system. The Figure Understander (Rajagopalan and Kuipers, 1994) was also developed to integrate text and diagram representations in the physics domain. Figure Understander was used to input problems into a magnetic fields problem solving system. The system relied on a system of figure semantics that related shading and patterns in the diagram with a semantic interpretation for the items. For example, a circle with white shading represented a loop of conducting wire, while black shading represented an immobile supporting object. In contrast to these approaches, MMKCap uses the more flexible concept labeling feature of CogSketch to allow users to enter objects and attach meaning without relying on a pre-defined library of mappings.

Watanabe and Nagao (1998) used a combination of spatial information and simple parsing rules to categorize the text associated with diagrams in a Japanese pictorial book of flora. Their method was specifically aimed at being able to classify the type of text (whether it described a plant species, plant part, etc) based on a combination of textual and spatial information and was limited to the domain of Japanese wild flowers. They also hand-coded the spatial relationships between text and diagrams. Currently MMKCap does not take advantage of this type of information, since CogSketch does not capture information about the placement of the conceptual label text. This suggests an interesting area of future work for our system, i.e., developing methods to capture the information that can be gleaned from the placement of a label in a textbook diagram.

The HALO project (Chaudhri *et al*, 2007; Clark *et al*, 2007) also addressed learning from textbooks and solving problems with the captured knowledge. The AURA system provided an interface for subject matter experts to input the knowledge from 50 pages of textbooks in each of physics, chemistry, and biology. The associated question answering system used a controlled language to allow

users to input AP-like test questions for the system to solve. The approach used in AURA focused on human-generated knowledge and on conceptual knowledge and tables (diagrams were excluded). This approach can be viewed as complementary to the work here as it uses subject-matter experts for the knowledge capture portion rather than an automated system.

MMKCap can also be viewed as a particular form of learning by reading. The closest systems are Mobius (Barker *et al*, 2007) and Learning Reader (Forbus *et al*, 2007). Mobius was used to see how existing NL and KR components could be combined to learn from text. It focused on two narrow domains (how human hearts and simple engines work), but was tested with a variety of paragraphs written by different people about those topics. Its knowledge base was small and hand-coded for the domain, and its learned knowledge was evaluated by hand inspection. Learning Reader, like MMKCap, uses simplified English and ResearchCyc KB contents, but a DMAP parser instead of the more traditional EA NLU system used here. Learning Reader was tested via automatically generated quizzes, and incorporated a process of rumination, where the system asked itself questions off-line to improve its performance later on. Both Mobius and Learning Reader were purely text-based, unlike the multimodal approach presented in this chapter. The system described here will be part of a next-generation learning by reading system, also incorporating ideas from Learning Reader.

6 CONCLUSIONS AND FUTURE WORK

6.1 DISCUSSION

This dissertation began by examining the problems of spatial preposition use and multimodal knowledge capture. While often approached as two distinct areas of research, the problems share several commonalities. Both multimodal knowledge capture and spatial preposition use are tasks that humans typically develop a robust ability for, but that remain challenging for Artificial Intelligence systems. One reason for this is that both tasks draw on a variety of competencies, including knowledge about objects in the world, spatial reasoning, and the ability to attend to shared relational structure. Spatial preposition use requires the user to be able to extract the important features from a visual scene based on previously learned categories. Multimodal knowledge capture requires the learner to integrate information from two distinct modalities based on coreference between the two sources. The reliance of these tasks on finding and exploiting common structure forms the basis for two claims:

- 1) Sequential generalization can be used to model the learning of spatial prepositions taking into account both functional and geometric features of a scene. In addition, sequential generalization can learn spatial preposition categories using far fewer training trials than existing models.
- 2) Structure mapping can be used to model the integration of multi-modal knowledge sources in a domain-general fashion without relying on predefined, domain-specific conventions.

The first claim is supported by three sets of experiments described in Chapter 4. The geometric shapes and cross-linguistic experiments show that sequential generalization (SEQL) can be used to model the use of spatial prepositions. The experiments used a variety of stimuli including geometric shapes in the geometric experiments and real-world objects in the cross-linguistic experiment. The cross-linguistic

experiment also showed that sequential generalization could be used to model spatial preposition use in two different languages. Not only can sequential generalization model spatial preposition use, a comparison with other computational models showed that it can do so using far fewer training examples than other computational models.

The second claim was explored in Chapter 5, where Mayer's multimedia learning theory formed the basis for the MMKCap model of multimodal knowledge capture which integrates text and diagram combinations. In MMKCap, text is first translated into controlled English and then processed using EA NLU which creates a discourse case containing predicate calculus facts representing the semantic content of the original text. Diagrams are sketched using CogSketch which then creates a diagram case containing predicate calculus facts about the objects in the sketch and the qualitative spatial relationships between them. The discourse and diagram cases are integrated using the Structure-Mapping Engine (SME) model of analogy. The MMKCap model was demonstrated using a chapter from *Basic Machines*, and evaluated using the publisher-provided homework questions for that chapter.

Chapter 5 also introduced some preliminary experiments on conceptual segmentation – the intelligent assignment of conceptual interpretations to the regions and edges in a sketch. This work showed that a combination of knowledge about an object and the ink, with which it is drawn, can be used to find the intended extent of that object in a sketched diagram.

This chapter describes some directions for future work in each of these areas. Section 6.2.1 discusses future directions for the spatial prepositions experiments of Chapter 4. Section 6.2.2 introduces ideas for the next steps in conceptual segmentation. Section 6.2.3 describes the future of MMKCap. Section 6.3 contains concluding remarks.

6.2 FUTURE WORK

6.2.1 FUTURE WORK IN SPATIAL PREPOSITIONS

The first goal of future work on spatial prepositions is to close the loop, so that the results from categorization experiments with SEQL can directly feed into the type of rules required for a SpaceCase-like system. This will allow the application of the learned, general-purpose preposition categories to other systems. Many applications using text, diagrams or a combination could be enhanced by adding a comprehensive understanding of spatial preposition use. This will be especially useful in applications where human users and AI agents need to collaborate around shared spatial artifacts (such as maps). Having a shared understanding of spatial language will allow AI systems and their human users to understand each other's intentions much more clearly. Such a system would also need to have the flexibility to continuously update its understanding of the preposition categories based on newly encountered uses and human feedback.

Developing an open-domain, flexible understanding of spatial preposition use will require a large number of labeled example sketches. The experiments in Chapter 4 focus on recreating results from psychological experiments. This is a good first step towards preposition understanding, but controlled experiments with tens of stimuli cannot truly capture the full range of situations in the real-world that humans encounter. A larger library of sketches will need to be developed to serve as a training set for future preposition work. These sketches can be drawn in part from other psychology work, but it would be interesting to branch out and consider other sources: children's books, textbooks, or even online sources. Along with having sketches labeled with English prepositions, larger corpuses labeled in different languages will be required to extend the cross-linguistic modeling. In addition to extending to more diagrams, SpaceCase should be able to operate on more complex diagrams. Moving

beyond simple, two object diagrams will require the system to be able to automatically identify an appropriate ground object.

6.2.2 FUTURE WORK IN CONCEPTUAL SEGMENTATION OF DIAGRAMS

The conceptual segmentation system in Chapter 5 was an early, proof-of-concept implementation, so there is a lot to tackle in future work. The first step is to use materials from the multimodal knowledge capture experiments to build up a larger corpus of sketched diagrams. A larger corpus will help provide a testbed for refining the conceptual segmentation algorithm. Also, more complex diagrams will likely involve drawing conventions, such as cut-aways or call-outs, that have different implications for how they should be parsed into conceptually meaningful parts.

In addition to multimodal knowledge capture, conceptual segmentation could be useful in a variety of other sketching tasks. Once the algorithm is refined, conceptual segmentation will be incorporated in future versions of the CogSketch system. There it would be applied to student worksheets and more general sketching applications. Moving beyond textbook diagrams means that conceptual segmentation will need to be robust to noise in sketches. For example, students may add unnecessary detail (like shading) to a sketch or misuse drawing conventions within a discipline. While situations such as these add challenges to conceptual segmentation, they also highlight the benefit of a flexible, domain-independent solution over strictly constrained recognition.

6.2.3 FUTURE WORK IN MULTIMODAL KNOWLEDGE CAPTURE

The first step in future work for MMKCap is to expand the number and types of sources processed. This will ensure that MMKCap is developed as a general multimodal knowledge capture tool, as opposed to becoming overly tailored to one type of source. In addition to other adult-level textbooks, there are plans to address learning materials designed for different audiences (such as

middle school science books). One difficulty in acquiring new source material is finding sources that contain a significant number of useful diagrams and also come with an external source of evaluation (like the publisher-provided assignments in *Basic Machines*). It might be interesting to test the ability of MMKCap to transfer knowledge from one source to the assignment from another. For example, knowledge learned from a physics textbook should not only lead to solving the assignments for that book, but also be useful for AP Physics test questions.

Automating several of the steps in MMKCap that require human intervention is another near-term goal. Using web-based sources with links and a tag structure may make the chunking of text and diagrams relatively easy to automate. Exploiting the structure of textbooks (i.e. subsections, chapter headings, etc) along with other conventions (such as introducing new concepts in bold or italic print) may make it possible to automate some of the knowledge engineering and also eliminate the need for manual disambiguation. Another goal is for the system to automatically identify new concepts and create collections for them, using the chapter structure to place the new concept in the hierarchy. For example, in the evaluation in Chapter 5, the concepts for first, second, and third class levers had to be created manually before the text was processed. The KB already contained the concept Lever. In the first chapter of *Basic Machines*, the first two sections on levers are “The Lever” and “Classes of Levers”. The “Classes of Levers” section has three subsections “First Class”, “Second Class” and “Third Class”. Exploiting this structure, the system could recognize that there were three new things being introduced, and that they were all types of levers. First, second and third class lever concepts could be created and added as a subclass of levers. Additionally, multi-word strings could be created for “first class lever”, etc.

A similar strategy could be used to aid in disambiguation. The system could keep a list of domain-specific vocabulary based on chapter and section headings and bolded/italicized words. Then,

when an ambiguous word is found, the available senses could be compared against the senses common in the running vocabulary list. For example, in *Basic Machines*, “gear” is an ambiguous word, there are two sense in the KB corresponding to the 1) a wheel with teeth and 2) personal gear (e.g. camping gear). The gens hierarchy in the KB could be used to determine that within the context of *Basic Machines*, it is much more likely that sense 1 was the intended meaning of “gear”.

While the majority of the diagrams in *Basic Machines* provided important information, not all textbook diagrams are equally useful for learning. Levin (1981) and others have examined what types of illustrations appear in textbooks and what types of diagrams or pictures lead to improved learning outcomes (e.g. Levin and Mayer, 1993). Some illustrations are simply decorative and have a minimal impact on learning. It would be interesting to see if the determination of the usefulness of a given diagram could be made automatically based on its characteristics. If the system could automatically identify which diagrams were necessary and which were superfluous it would reduce the chance that distracting information gets captured.

Another area that requires additional work is the encoding of both worked examples and procedural knowledge needed for problem solving. The translation of a worked example in the text to the predicate calculus representation of problem solving steps is the first step in this process. Slightly less straightforward is the capture and representation of the metacognitive knowledge about how and when to employ different problem solving strategies. The evaluation in Chapter 5 showed that often basic transfer from worked example to homework problem would miss critical problem solving steps. The goal is to capture this kind of information in a general purpose way so that the system can build a repertoire of problem solving techniques, applicable across different sources.

6.3 CONCLUSION

This dissertation has examined the application of analogy and qualitative spatial representation to models of spatial preposition use and multimodal knowledge capture. In Chapter 4, it was shown that sequential generalization over sketched inputs can be used to model the learning of spatial prepositions. This process was shown to be effective over sketches of varying complexity (simple shapes versus real world objects) and in two languages. In addition, the SpaceCase model was introduced. SpaceCase uses evidential rules to assign a spatial preposition to a novel visual scene.

Chapter 5 presented the MMKCap model of multimodal knowledge capture. MMKCap uses CogSketch to create a representation from a textbook diagram, and EA NLU to extract the semantic meaning from the associated text. Then, SME is used to integrate the two representations, showing that structure-mapping can be used to model integration of multimodal knowledge sources. The MMKCap model was demonstrated using a chapter from *Basic Machines* and was evaluated using the publisher-provided homework for that chapter.

There were three main bodies of work in this dissertation: (1) spatial prepositions, (2) conceptual segmentation of diagrams and (3) multimodal knowledge capture. Each has its own possibilities for future work described in this chapter. However, some of the most interesting possibilities may come from combining these models. For example, can a better understanding of spatial prepositions lead to more natural conceptual segmentation? How can conceptual segmentation be used in MMKCap?

7 WORKS CITED

- Allen, J.F. (1995). *Natural Language Understanding* (2nd ed). Redwood City, CA: Benjamin/Cummings.
- Alvarado, C., Oltmans, M., and Davis, R. (2002). A framework for multi-domain sketch recognition. In *Proceedings of AAAI Spring Symposium on Sketch Understanding*.
- Alvarado, C., Davis, R. (2004). Sketchread: a multi-domain sketch recognition engine. *Proceedings of the 17th annual ACM symposium on user interface software and technology*.
- Anderson, M. and Armen, C. (2002). DiaSketches. *Proceedings of the 2nd international symposium on Smart graphics*, 55 – 62.
- Andre, E., Herzog, G. and Rist, T. (1988). On the simultaneous interpretation of real world image sequences and their natural language description: The system SOCCER. In *Proceedings of the 8th ECAI*(pp, 449-454). London: Pittman.
- Barker, K. et al. (2007). Learning by Reading: A Prototype System, Performance Base-line, and Lessons Learned. *AAAI'07*.
- Basic Machines*. (1994). Navy Education and Training Professional Development and Technology Center.
- Blocher, A. and Stopp, E. (1998). Time-Dependent Generation of Minimal Sets of Spatial Descriptions. In *Representation and Processing of Spatial Expressions* (eds. Oliver, P and Gapp, K-P.) Lawrence Erlbaum Associates, Mahwah, NJ. 57-71.
- Bowerman, M. and Pederson, E. (In preparation). INwards from ON and ONwards from IN: The crosslinguistic categorization of topological spatial relationships.
- Bowerman, M. (1996). Learning how to structure space for language: A crosslinguistic perspective. In P. Bloom, M.A. Peterson, L. Nadel, and M.F. Garrett (Eds.), *Language and Space* (pp. 384-436). Cambridge, MA: MIT Press.
- Bowerman, M. and Choi, S. (2001). Shaping Meanings for Language: Universal and Language-Specific in the acquisition of spatial semantic categories. In M. Bowerman and S. Levin (Eds.) *Language Acquisition and Conceptual Development*, 475-511. Cambridge: Cambridge University Press.
- Brugman, C. (1981). *The story of "over"*. Unpublished master's thesis, University of California, Berkeley.
- Brugman, C. (1988). *The story of "over": Polysemy, semantics and the structure of the lexicon*. Garland Press.
- Brugman, C. and Lakoff, G. (1988). Cognitive topology and lexical networks. In G.W. Cottrell, S.Small, & M.K. Tannenhouse (Eds.), *Lexical ambiguity resolution: Perspectives from psycholinguistics, neuropsychology, and artificial intelligence*. San Mateo, CA: Morgan Kaufman.

- Buckley, S. (1979). *Sun Up to Sun Down*. New York: McGraw Hill.
- Bulko, W.C. (1988). Understanding Text with an Accompanying Diagram. *Proceedings of the 1st International Conference on Industrial Engineering Applications of Artificial Intelligence and Expert Systems*, 894-898.
- Cangelosi A., K. Coventry, R. Rajapakse, D. Joyce, A. Bacon, L. Richards, S. Newstead (2005), Grounding language in perception: A connectionist model of spatial terms and vague quantifiers. In A. Cangelosi, G. Bugmann & R. Borisyuk (Eds.), *Modelling Language, Cognition and Action: Proceedings of the 9th Neural Computation and Psychology Workshop*, 47-56.
- Carlson-Radvansky, L. A. and Radvansky, G. A. (1996). The influence of functional relations on spatial term selection. *Psychological Science* 7, 56–60.
- Carlson-Radvansky, L. A., Covey, E. S., and Lattanzi, K. M. (1999). "What" effects on "where": Functional influences on spatial relations. *Psychological Science*, 10, 516-521.
- Carlson-Radvansky, L.A. and Logan, G.D. (1997). The Influence of Reference Frame Selection on Spatial Template Construction. *Journal of Memory and Language* 37(3), Pages 411-437.
- Chaudhri, V. et al. (2007). Enabling Experts to Build Knowledge Bases from Science Textbooks. *KCap'07*.
- Clark, P. et al. (2007). Capturing and Answering Questions Posed to a Knowledge-Based System. *KCap'07*.
- Cohn, A. (1996). Calculi for Qualitative Spatial Reasoning. In *Artificial Intelligence and Symbolic Mathematical Computation*, LNCS 1136, eds: J Calmet, J.A. Cambell, J Pfalzgraph. Springer Verlag, 124-143.
- Cooper, G.S. (1968). *A Semantic Analysis of English Locative Prepositions*. Springfield, VA: Clearinghouse for Federal Scientific and Technical Information.
- Coventry, K. R., Carmichael, R., and Garrod, S. C. (1994). Spatial prepositions, object-specific function and task requirements. *Journal of Semantics*, 11, 289-309.
- Coventry, K. R., Cangelosi, A., Rajapakse, R., Bacon, A., Newstead, S., Joyce, D. & Richards, L. V. (2005). Spatial prepositions and vague quantifiers: Implementing the functional geometric framework. In C. Freksa, B. Knauff & B. Krieg-Bruckner & B. Nebel (Eds.), *Spatial Cognition, Volume IV. Reasoning, Action and Interaction*, 98-110.
- Coventry, K. R., (1998). Spatial prepositions, functional relations and lexical specification. In P. Olivier & K. Gapp (Eds.), *The Representation and Processing of Spatial Expressions*. Mahwah, NJ: Lawrence Erlbaum Associates, 247-262.
- Coventry, K. R. (1999). Function, geometry and spatial prepositions: Three experiments. *Spatial Cognition and Computation*, 2, 145-154.

- Coventry, K. R., Prat-Sala, M., and Richards, L. V. (2001). The interplay between geometry and function in the comprehension of "over", "under", "above" and "below". *Journal of Memory and Language*, 44, 376-398.
- Coventry, K. and Mather, G., (2002). The real story of *Over*. In P. Olivier and W. Maass (Eds.), *Representations between vision and language*. New York:Springer-Verlag.
- Coventry, K. and Garrod, S. (2004). *Seeing, Saying and Acting: The psychological semantics of spatial prepositions*. Hove: Psychology Press.
- Dehghani, M., Tomai, E., Forbus, K. and Klenk, M. (2008). An Integrated Reasoning Approach to Moral Decision-Making. In the *Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI)*. Chicago, IL.
- DiTomaso, V., Lombardo, V., and Lesmo, L. (1998). A Computational Model for the Interpretation of Static Locative Expressions. In *Representation and Processing of Spatial Expressions* (eds. Oliver, P and Gapp, K-P.) Lawrence Erlbaum Associates, Mahwah, NJ. 73-90.
- Edwards, G. and Moulin, B. (1998). Toward the Simulation of Spatial Mental Images Using the Voronoi Model. In *Representation and Processing of Spatial Expressions* (eds. Oliver, P and Gapp, K-P.) Lawrence Erlbaum Associates, Mahwah, NJ. 163-184.
- Everett, J.O. (1999). Topological Inference of Teleology: Deriving Function from Structure via Evidential Reasoning. *Artificial Intelligence* 113 (1-2).
- Falkenhainer, B. (1988). Learning from Physical Analogies. Technical Report No. UIUCDCS-R-88-1479, University of Illinois at Urbana-Champaign. (Ph.D. Thesis)
- Falkenhainer, B., Forbus, K., and Gentner, D. (1989). The Structure-Mapping Engine: Algorithms and Examples. *Artificial Intelligence*, 41, 1-63.
- Feist, M. I., & Gentner, D. (1998). On plates, bowls, and dishes: Factors in the use of English IN and ON. *Proceedings of the Twentieth Annual Meeting of the Cognitive Science Society*, 345-349.
- Feist, M.I., & Gentner, D. (2001). An influence of spatial language on recognition memory for spatial scenes. *Proceedings of the Twenty-Third Annual Meeting of the Cognitive Science Society*.
- Feist, M.I., & Gentner, D. (2003). Factors involved in the use of in and on. *Proceedings of the Twenty-Fifth Annual Meeting of the Cognitive Science Society*.
- Ferguson, R., Rasch, R.A., Turmel, W. and Forbus, K. (2000). Qualitative spatial interpretation of Course-of-Action diagrams. In *Proceedings of the 14th International Workshop on Qualitative Reasoning*. Morelia, Mexico.

- Ferguson, R.W. & Forbus, K. (1995). Under-standing Illustrations of Physical Laws by Integrating Differences in Visual and Textual Representations. *AAAI Fall Symposium on Computational Modeling for Integrating Language and Vision*.
- Forbus, K., Riesbeck, C., Birnbaum, L., Livingston, K., Sharma, A., & Ureel, L. (2007). A Prototype System that Learns by Reading Simplified Texts. *AAAI Spring Symposium on Machine Reading*. Stanford University, California.
- Forbus, K. (1984). Qualitative process theory. *Artificial Intelligence*, 24, 85-168.
- Forbus, K. and deKleer, J. (1993). *Building Problem Solvers*. MIT Press.
- Forbus, K., Usher, J., Lovett, A., Lockwood, K. and Wetzel, J. (2008). CogSketch:Open-domain sketch understanding for cognitive science research and for education. *Proceedings of the Fifth Eurographics Workshop on Sketch-Based Interfaces and Modeling*.
- Forbus, K., Ferguson, R. and Usher, J. (2001). Towards a Computational Model of Sketching. In *Intelligent User Interfaces (IUI)*.
- Forbus, K., Gentner, D. and Law, K. (1995). MAC/FAC: A model of Similarity-based Retrieval. *Cognitive Science*, 19(2), 141-205.
- Friedman, S. and Forbus, K. (2008). Learning Causal Models via Progressive Alignment & Qualitative Modeling: A Simulation. In the *Proceedings of the 30th Annual Conference of the Cognitive Science Society (CogSci)*. Washington, DC.
- Fuhur, T., Socher, G., Scheering, C. and Sagerer, G. (1998). A Three-Dimensional Spatial Model for the Interpretation of Image Data. In *Representation and Processing of Spatial Expressions* (eds. Oliver, P and Gapp, K-P.) Lawrence Earlbaum Associates, Mahwah, NJ. 103-118.
- Futrelle, R. P. (1990). Strategies for Diagram Understanding: Object/Spatial Data Structures, Animate Vision and Generalized Equivalence. In *10th ICPR* (pp. 403-408): IEEE Press.
- Gapp, K-P. (1995a). Angle, Distance, Shape, and their Relationship to Projective Relations. *Proceedings of the Seventeenth Annual Meeting of the Cognitive Science Society*.
- Gapp, K-P. (1995b). An Empirically Validated Model for Computing Spatial Relations. *Proceedings of the Nineteenth Annual German Conference on Artificial Intelligence*.
- Garrod, S., Ferrier, G.& Campbell, S. (1999) In and On: Investigating the functional geometry of spatial prepositions. *Cognition*, 72, 167-189
- Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7, 155-170.

- Gentner, D. and Bowerman, M. (2009). Why Some Spatial Semantic Categories are Harder to Learn than Others: The Typological Prevalence Hypothesis. In (Guo, J., Lieven, E., Budwig, N., Ervin-Tripp, S., Nakamura, K., and Ozxaliskan, S. Eds). *Crosslinguistic Approaches to the Psychology of Language: Research in the Tradition of Dan Isaac Slobin*. CRC Press.
- Halstead, D. and Forbus, K. (2007). Some Effects of a Reduced Relational Vocabulary in the Whodunit Problem. *Proceedings of IJCAI-2007*, Hyderabad, India.
- Halstead, D. and Forbus, K. (2005). Transforming between Propositions and Features: Bridging the Gap. *Proceedings of AAAI*, Pittsburg, PA.
- Harp, S. F. and Mayer, R.E. (1997). The role of interest in learning from scientific text and illustrations : On the distinction between emotional interest and cognitive interest. *Journal of Educational Psychology*, 89(1), 92-102.
- Hegarty, M. & Just, M. A. (1989). Understanding machines from text and diagrams. In H. Mandl & J. Levin (Eds.). *Knowledge acquisition from text and picture*. Amsterdam: North Holland.
- Hegarty, M. & Just, M.A. (1993). Constructing mental models of machines from text and diagrams. *Journal of Memory and Language*, 32, 717-742
- Herskovits, A. (1985). Semantics and pragmatics of spatial prepositions. *Cognitive Science*, 9, 341-378.
- Herskovits, A. (1986). *Language and Spatial Cognition: an interdisciplinary study of the prepositions in English*. Cambridge, England: Cambridge University Press.
- Herskovits, A. (1998). Schematization. In P. Oliver and K. Gapp (Eds.), *Representation and Processing of Spatial Expressions*. Hillsdale, NJ:L. Erlbaum Associates, 149-162.
- Kamp, H. and Reyle, U. (1993). *From Discourse to Logic*. Kluwer, Dordrecht.
- Kara, L. B. and Stahovich, T. F. (2004). Hierarchical parsing and recognition of hand-sketched diagrams. *Proceedings of the 17th annual ACM symposium on User interface software and technology*, 13 – 22.
- Klenk, M. and Forbus, K. (2007). Cognitive modeling of analogy events in physics problem solving from examples. In the *Proceedings of CogSci-2007*. Nashville, TN.
- Kuehne, S., Gentner, D. and Forbus, K. (2000). Modeling infant learning via symbolic structural alignment. *Proceedings of CogSci 2000*.
- Kuehne, S. and Forbus, K. (2004). Capturing QP-Relevant Information from Natural Language Text. *Proceedings of QR04*.
- Kuehne, S., Forbus, K., Gentner, D. and Quinn, B. (2000). SEQL: Category learning as progressive abstraction using structure mapping. In *Proceedings of the 22nd Annual Meeting of the Cognitive Science Society*.

- Landau, B. and Jackendoff, R. (1993). 'What' and 'where' in spatial language and spatial cognition. *Behavioral and Brain Sciences*, 16, 217-265.
- Landau, B. (1996). Multiple geometric representations of objects in languages and language learners. In P. Bloom, M.A. Peterson, L. Nadel, and M.F. Garrett (Eds.), *Language and Space* (pp. 317-365). Cambridge, MA: MIT Press.
- Lakoff, G. (1987). *Women, Fire, and Dangerous Things*. Chicago: University of Chicago Press.
- Larkin, J. & Simon, H. (1987) Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science*, 11, 65-99
- Lenat, D. B. (1995). CYC: A Large-Scale Investment in Knowledge Infrastructure. *Communications of the ACM*. 38(11), 33-38.
- Levie, W.H. and Lentz, R. (1982). Effects of text illustrations: A review of research. *Educational Communication and Technology Journal*, 30, 195-232.
- Levin, J. (1981). On Functions of Pictures in Prose. In Pirozzolo, F.J. and Wittrock, M.C. (eds) *Neuropsychological and Cognitive Processes in Reading*. New York: Academic Press.
- Levin, J. and Mayer, R. (1993). Understanding Illustrations in Text. In Britton, B., Woodward, A., and Binkley, M. (eds) *Learning from textbooks: Theory and Practice*. Lawrence Erlbaum.
- Lockwood, K. et al (2008). A Theory of Depiction for Sketches of Physical Systems. *Proceedings of QR08*.
- Lockwood, K., Forbus, K., Halstead, D. & Usher, J. (2006). Automatic Categorization of Spatial Prepositions. *Proceedings of the 28th Annual Conference of the Cognitive Science Society*. Vancouver, Canada.
- Lockwood, K., Forbus, K., & Usher, J. (2005). SpaceCase: A model of spatial preposition use. *Proceedings of the 27th Annual Conference of the Cognitive Science Society*. Stressa, Italy.
- Logan, G. D. and Sadler, D. D. (1996). A computational analysis of the apprehension of spatial relations. In P. Bloom, M. A. Peterson, L. Nadel and M. Garrett (Eds.). *Language and space*. Cambridge, MA: MIT Press , pp. 493–529.
- Lovett, A., Dehghani, M. and Forbus, K. (2008). Building and comparing qualitative descriptions of three-dimensional design sketches. In *Proceedings of the 22nd International Qualitative Reasoning Workshop*. Boulder, CO.
- Lovett, A., Lockwood, K., Dehghani, M. and Forbus, K. (2007). Modeling human-like rates of learning via analogical generalization. *Proceedings of Analogies: Integrating Multiple Cognitive Abilities*. Nashville, Tennessee.

- Lovett, A., Lockwood, K. and Forbus, K. (2008). A computational model of the visual oddity task. In the *Proceedings of the 30th Annual Conference of the Cognitive Science Society*. Washington, DC.
- Macleod, C., Grashman, R. and Meyers, A. (1998). COMLEX Syntax Reference Manual, Version 3.0. Linguistic Data Consortium. University of Pennsylvania: Philadelphia, PA.
- Matuszek, C., Cabral, J., Witbrock, M. and DeOliveira, J. (2006). An Introduction to the Syntax and Content of Cyc. In *Proceedings of the 2006 AAAI Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*, Stanford, CA.
- Mayer, R. E. (1989). Systematic thinking fostered by illustrations in scientific text. *Journal of Educational Psychology*, 81, 240-246.
- Mayer, R. E. (2001). *Multimedia Learning*. Cambridge: Cambridge University Press.
- Mayer, R. E. and Gallini, J. (1990). When is an illustration worth ten thousand words? *Journal of Educational Psychology*, 82, 715-726.
- Mayer, R. E. and Simms, V. (1994). For Whom is a Picture Worth a Thousand Words? Extensions of a Dual-Coding Theory of Multimedia Learning. *Journal of Educational Psychology* 86(3), 389-401.
- Mayer, R. E., Steinhoff, K., Bower, G., and Mars, R. (1995). A generative theory of textbook design: Using illustrations to foster meaningful learning of science text. *Educational Technology Research & Development*, 43, 31-43.
- Mayer, R.E., Bove, W., Bryman, A., Mars, R., and Tapangco, L. (1996). When less is more: Meaningful learning from visual and verbal summaries of science textbook lessons. *Journal of Educational Psychology*, 82, 64-73.
- McDonough, L, Choi, S., and Mandler, M. (2003). Understanding spatial relations: Flexible infants, lexical adults. *Cognitive Psychology*, 46(3), 229-259.
- Miller, G.A. and Johnson-Laird, P.N. (1976). *Language and Perception*. Cambridge, MA: Harvard University Press.
- Mukerjee, A. (1998). Neat Versus Scruffy: A Review of Computational Models for Spatial Expressions. In P. Oliver and K. Gapp (Eds.), *Representation and Processing of Spatial Expressions*. Hillsdale, NJ:L. Erlbaum Associates, 1-35.
- Paivio, A. (1986). *Mental Representations: A Dual Coding Approach*. Oxford, UK: Oxford University Press.
- Pearl, J. (1986). On Evidential Reasoning in a Hierarchy of Hypotheses. *Artificial Intelligence* 28(1), 9-15.
- Peeck, J. (1989). Trends in the delayed use of information from an illustrated text. In H. Mandl & J. Levin (Eds.). *Knowledge acquisition from text and picture*. Amsterdam: North Holland.

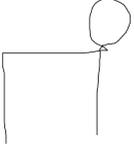
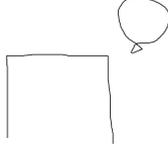
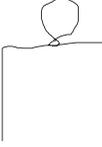
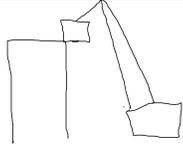
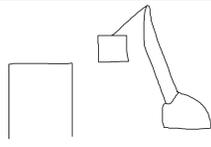
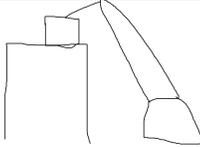
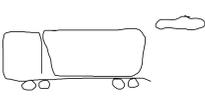
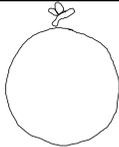
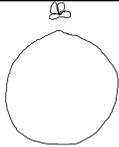
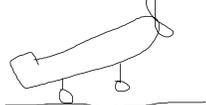
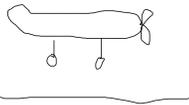
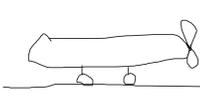
- Rajagopalan, R. & Kuipers, B. (1994). The Figure Understander: A system for integrating text and diagram input to a knowledge base. *Proceedings of the 7th Inter-national Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*.
- Regier, T. (1995). A Model of the Human Capacity for Categorizing Spatial Relations. *Cognitive Linguistics*, 6(1), 63-88.
- Regier, T. (1996). *The Human Semantic Potential: Spatial Language and Constrained Connectionism*. MIT Press.
- Regier, T. & Carlson, L. A. (2001). Grounding spatial language in perception: An empirical and computational investigation. *Journal of Experimental Psychology: General*, 130, 273-298
- Regier, T., Carlson, L. A., & Corrigan, B. (2004). Attention in spatial language: Bridging geometry and function. In L. Carlson & E. van der Zee (Eds.). *Functional features in language and space: Insights from perception, categorization and development*. Oxford University Press.
- Reiter, R. and Mackworth, A.K. (1989). A Logical Framework for Depiction and Image Interpretation. *Artificial Intelligence*, 41, 125-155.
- Saund, E., Mahoney, J., Fleet, D., Lerner, D., and Lank, E. (2002). Perceptual Organization as a Foundation for Intelligent Sketch Editing. In *Proc AAAI Spring Symposium on Sketch Understanding*.
- Saund, E. (2003). Finding Perceptually Closed Paths in Sketches and Drawings. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 25:4, 475-491.
- Searle, J. R. (1979). Literal Meaning. *Expression and Meaning*. Cambridge, England: Cambridge University Press.
- Skorstad, J., Gentner, D. and Medin, D. (1998). Abstraction Process During Conceptual Learning: A Structural View. *Proceedings of the 10th Annual Conference of the Cognitive Science Society*.
- Talmy, L. (1983). How language structures space. In H. Pick and L. Acredolo (eds.), *Spatial orientation: Theory, research, and application*. New York: Plenum, 225-282.
- Tomai, E. and Forbus, K. (2009). EA NLU: Practical Language Understanding for Cognitive Modeling. *Proceedings of the 22nd International Florida Artificial Intelligence Research Society Conference*. Sanibel Island, Florida.
- Watanabe, Y. & Nagao, M. (1998). Diagram Understanding using Integration of Layout Information and Textual Information. *Proceedings of the Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*.

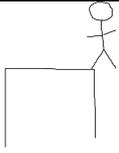
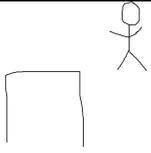
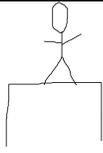
Wetzel, J. and Forbus, K. (2008). Integrating Open-Domain Sketch Understanding with Qualitative Two-Dimensional Rigid-Body Mechanics. In the *Proceedings of the 22nd International Workshop on Qualitative Reasoning*. Boulder, CO.

8 APPENDICES

Appendices for this dissertation appear on the following pages.

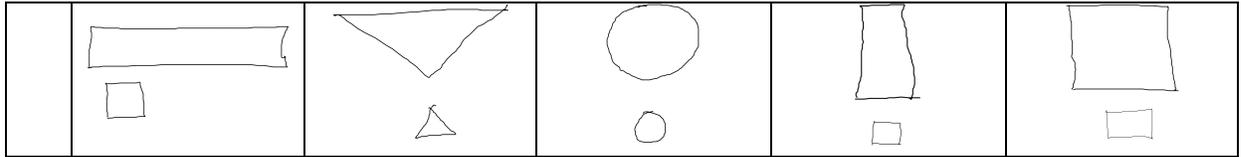
8.1 APPENDIX A: STIMULI FOR SPACECASE EXPERIMENT #2: RETRIEVAL

Initial Variant	Minus Variant	Plus Variant
		 balloon <i>on</i> stick
		 balloon <i>on</i> table
		 block <i>on</i> building
		 coin <i>in</i> hand
		 dirt <i>in</i> truck
		 firefly <i>on</i> dish
		 firefly <i>on</i> wheel
		 plane <i>on</i> ground

		 <p data-bbox="1144 420 1339 451">puppet <i>on</i> table</p>
		 <p data-bbox="1144 609 1339 640">spider <i>on</i> bowl</p>

8.2 APPENDIX B: STIMULI FOR GEOMETRIC EXPERIMENT 1

Simple in					
Simple left					
Simple on					
Simple over					
Simple					



8.3 APPENDIX C: FACTS FILTERED FROM THE SKETCH CASES

facts in **initial-space-filter** were filtered if they appeared in the first position in a statement, **third-isa-atom** facts were filtered if the listed collection appeared as the second argument in a statement with *isa* as the predicate.

<pre>(defparameter *initial-space-filter* 'd::(askConceptualForBinaryVisualRelation bboxLastModifiedTime containedGlyphGroupTangentialInsider directionalSignature entityTypesLastModifiedTime glyphGraphCWA glyphGraphEdgesFor glyphRepresentsObject hasRCC8Relation inkLastModifiedTime kbDateModified nameString nameStringLastModifiedTime needVisualPositionalRelation nuSketchCaseID nuSketchCreationMachine nuSketchCreator sketchCreatedWithVersion sketchFor sketchModifiedWithVersion sketchRepresentsObject subSketchFor subSketchGroupFor subSketchGroupRepresentsObject subSketchHasPose subSketchHasGenre subSketchRepresentsObject q-2D-orientation q-roundness userCWA voronoiFor))</pre>	<pre>(defparameter *third-isa-atom-filter* 'd::(Case Circle Entity Ellipse Glyph Individual LargeSizeGlyph LookingFromSide-SubSketch NotVeryRoundGlyph NuSketchBundle NuSketchCase NuSketchGlyph NuSketchLayer NuSketchSketch PhysicalView-SubSketch QDiagonalDownwardGlyph QDiagonalUpwardGlyph QHorizontalGlyph QVerticalGlyph Rectangle SmallSizeGlyph SomewhatRoundGlyph Sketch-Drawing StaticSituation Square SubSketch SubSketchGroup Triangle TwoDimensionalGeometricThing VeryRoundGlyph))</pre>
---	--

8.4 APPENDIX D: GENERALIZATIONS CREATED IN SIMPLE GEOMETRIC EXPERIMENT 1

Best Generalization (GenFn SpaceSeq|2 45) – EQUIVALENT TO *IN*

Size: 10

--DEFINITE FACTS:

(spatiallyIntersects (GlyphFn :genent0 :genent1) (GlyphFn :genent2 :genent1))

(ContainedGroup figure ground)

(subCaseOf :genent3 :genent4)

--POSSIBLE FACTS:

80%: (rcc8-NTPP figure ground)

20%: (rcc8-TPP figure ground)

--UNLIKELY FACTS:

Best Generalization (GenFn SpaceSeq|2 36) – EQUIVALENT TO *LEFT*

Size: 10

--DEFINITE FACTS:

(enclosesVertically ground figure)

(rcc8-DC figure ground)

(leftOf figure ground)

(subCaseOf :genent0 :genent1)

--POSSIBLE FACTS:

--UNLIKELY FACTS:

Best Generalization (GenFn SpaceSeq|2 27) – EQUIVALENT TO *ON*

Size: 10

--DEFINITE FACTS:

(rcc8-EC figure ground)

(enclosesHorizontally ground figure)

(connectedGlyphGroupTangentialConnection figure ground nil)

(connectedGlyphGroupTangentialConnection ground figure nil)

(above figure ground)

(ConnectedGroup figure ground)

(subCaseOf :genent0 :genent1)

--POSSIBLE FACTS:

--UNLIKELY FACTS:

Best Generalization (GenFn SpaceSeq|2 18) – EQUIVALENT TO *OVER*

Size: 10

--DEFINITE FACTS:

(aboveGrazingLine figure ground)

(enclosesHorizontally ground figure)

(above figure ground)

(rcc8-DC figure ground)

(subCaseOf :genent0 :genent1)

--POSSIBLE FACTS:

--UNLIKELY FACTS:

Best Generalization (GenFn SpaceSeq12 9) – EQUIVALENT TO *UNDER*

Size: 10

--DEFINITE FACTS:

(belowGrazingLine figure ground)

(below figure ground)

(enclosesHorizontally ground figure)

(rcc8-DC figure ground)

(subCaseOf :genent0 :genent1)

--POSSIBLE FACTS:

--UNLIKELY FACTS:

8.5 APPENDIX E: GENERALIZATIONS CREATED BY GEOMETRIC SHAPES EXPERIMENT 2

Best Generalization - *IN*

Size: 17

--DEFINITE FACTS:

(spatiallyIntersects (GlyphFn :genent0 :genent1) (GlyphFn :genent2 :genent1))

--POSSIBLE FACTS:

59%: (ContainedGroup figure ground)

47%: (rcc8-NTPP figure ground)

--UNLIKELY FACTS:

6%: (ConnectedGroup figure ground)

6%: (above figure ground)

6%: (enclosesHorizontally ground figure)

6%: (rcc8-PO figure ground)

6%: (definiteOverlapCase figure ground)

6%: (enclosesHorizontally figure ground)

6%: (enclosesHorizontally ground figure)

6%: (leftOf figure ground)

6%: (ConnectedGroup figure ground)

6%: (above figure ground)

6%: (rcc8-PO figure ground)

6%: (definiteOverlapCase figure ground)

Best Generalization - *ON*

Size: 17

--DEFINITE FACTS:

(rcc8-EC figure ground)

(connectedGlyphGroupTangentialConnection ground figure nil)

(connectedGlyphGroupTangentialConnection figure ground nil)

(ConnectedGroup figure ground)

--POSSIBLE FACTS:

88%: (above figure ground)

65%: (enclosesHorizontally ground figure)

--UNLIKELY FACTS:

6%: (enclosesHorizontally figure ground)

6%: (enclosesHorizontally figure ground)

6%: (enclosesVertically ground figure)

6%: (leftOf figure ground)

6%: (rightOf figure ground)

Best Generalization - *OVER*

Size: 13

--DEFINITE FACTS:

(aboveGrazingLine figure ground)

(above figure ground)

(rcc8-DC figure ground)
 --POSSIBLE FACTS:
 77%: (enclosesHorizontally figure ground)
 --UNLIKELY FACTS:
 8%: (enclosesHorizontally figure ground)
 8%: (leftOf figure ground)
 8%: (enclosesHorizontally figure ground)

Best Generalization – *LEFT OF*

Size: 10

--DEFINITE FACTS:
 (enclosesVertically figure ground)
 (leftOf figure ground)
 (rcc8-DC figure ground)
 --POSSIBLE FACTS:
 --UNLIKELY FACTS:

Best Generalization - *UNDER*

Size: 10

--DEFINITE FACTS:
 (belowGrazingLine figure ground)
 (below figure ground)
 (enclosesHorizontally figure ground)
 (rcc8-DC figure ground)
 --POSSIBLE FACTS:
 --UNLIKELY FACTS:

Best Generalization – *Ambiguous Over/Left of*

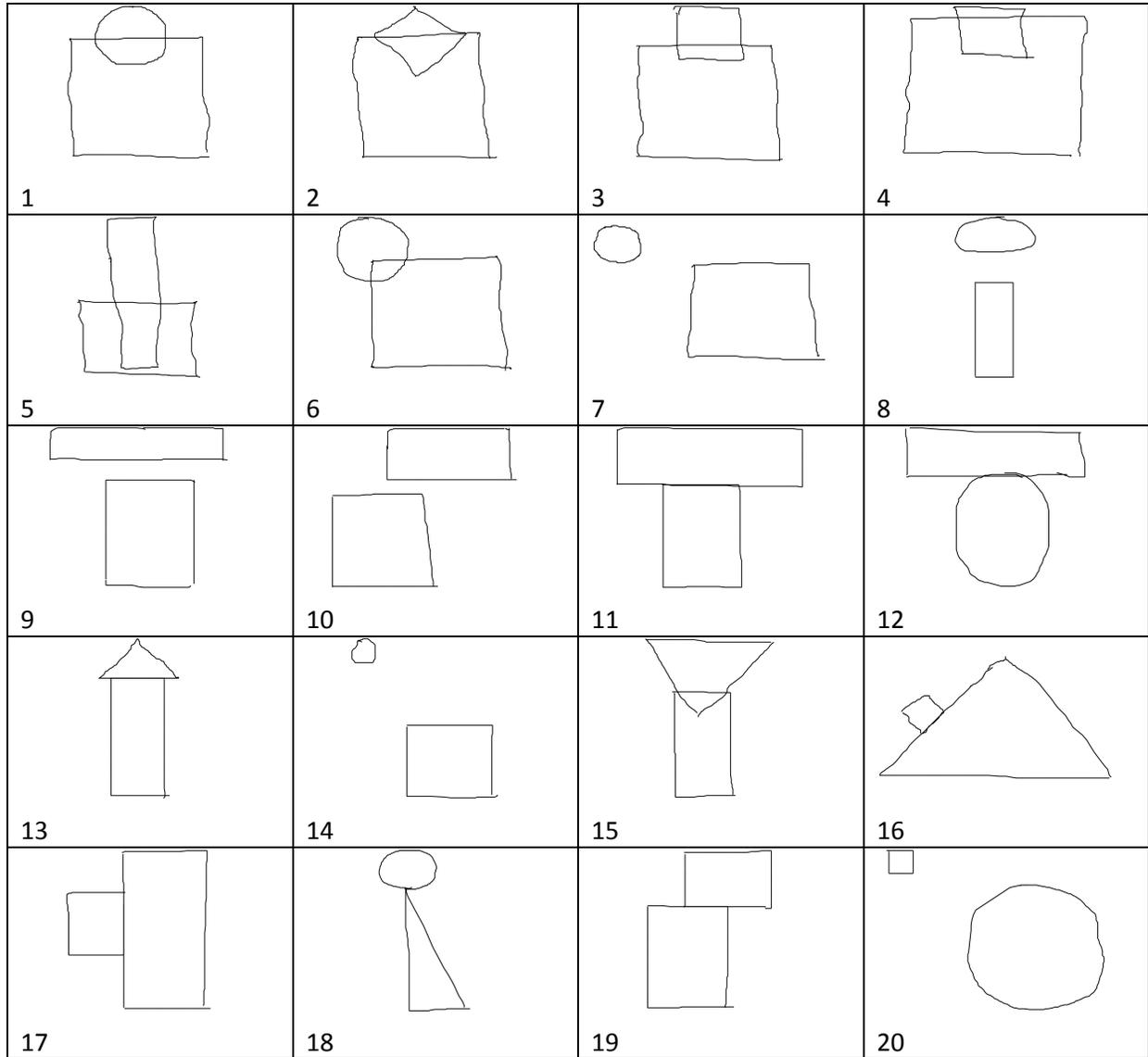
Size: 2

--DEFINITE FACTS:
 (leftOf figure ground)
 (rcc8-DC figure ground)
 (above figure ground)
 --POSSIBLE FACTS:
 --UNLIKELY FACTS:

Exemplar

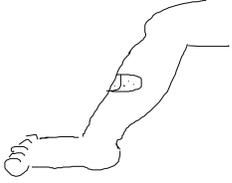
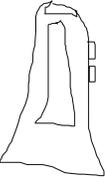
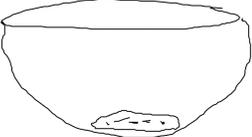
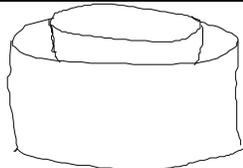
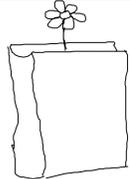
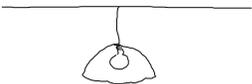
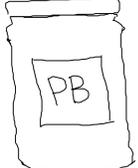
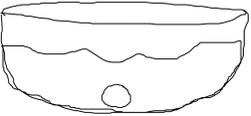
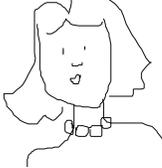
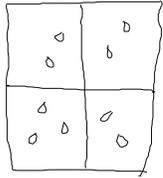
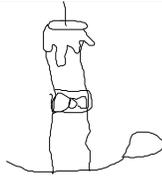
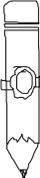
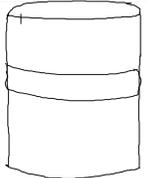
(rcc8-DC figure ground)
 (above figure ground)
 (rightOf figure ground)

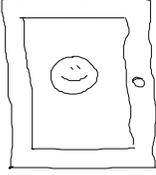
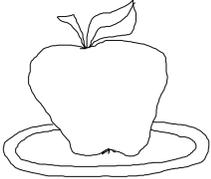
8.6 APPENDIX F: SIMPLE GEOMETRIC EXPERIMENT 2 STIMULI



8.7 APPENDIX G: STIMULI FOR EXPERIMENT # LEARNING SPATIAL PREPOSITIONS IN ENGLISH AND IN DUTCH

These sketches are all drawn from the stimuli used in Gentner and Bowerman (in press) and were sketched using the CogSketch system

			
bandaid <i>op</i> leg	bandana <i>om</i> head	button <i>aan</i> jacket	candle <i>in</i> bottle
			
clothes <i>aan</i> line	cookie <i>in</i> bowl	cookie <i>op</i> plate	cup <i>in</i> tube
			
flower <i>in</i> book	freckles <i>op</i> face	handle <i>aan</i> pan	hole <i>in</i> towel
			
hoop <i>om</i> doll	knob <i>aan</i> door	lamp <i>aan</i> ceiling	lid <i>op</i> jar
			
marble <i>in</i> water	mirror <i>aan</i> wall	necklace <i>om</i> neck	purse <i>aan</i> hook
			
raindrops <i>op</i> window	ribbon <i>om</i> candle	ring <i>om</i> pencil	rubber band <i>om</i> can

 <p>sticker <i>op</i> cupboard</p>	 <p>toy dog <i>op</i> book</p>	 <p>sick <i>in</i> straw</p>	 <p>top <i>op</i> tube</p>
 <p>tube <i>om</i> stick</p>	 <p>string <i>aan</i> balloon</p>	 <p>wrapper <i>om</i> gum</p>	 <p>apple <i>in</i> ring</p>

8.8 APPENDIX H: CONCEPTUAL LABELS USED IN THE CROSS-LINGUISTIC EXPERIMENT

Figure Objects		Ground Objects	
Original	KB Concept	Original	KB Concept
cookie	Cookie	plate	DinnerPlate
toy dog	Toy	book	BookCopy
bandaid	BandAidBandageProduct	leg	Leg
raindrops	Raindrop	window	WindowThePortalCovering
sticker	Sticker-Adhesive	cupboard	Cupboard
lid	Covering-Object	jar	Jar
top	Covering-Object	tube	Tube-Container
freckles	Freckle	face	FaceOfAnimal
mirror	Mirror-Wall	wall	WallOfAConstruction
purse	Purse	hook	Hook
clothes	Clothing-Generic	line	ClothesLine
lamp	Lamp-Hanging	ceiling	CeilingOfARoom
handle	Handle	pan	CookingVessel
string	String-Textile	balloon	Balloon
knob	DoorKnob	door	DoorInABuilding
button	ButtonTheFastener	jacket	Coat
necklace	Necklace	neck	Head-AnimalBodyPart
rubber band	RubberBand	can	Can
bandana	Bandana	head	Head-AnimalBodyPart
hoop	Circle	doll	Doll-Toy
ring	Ring-Jewelry	pencil	Pencil
tube	Tube	stick	TreeBranch
wrapper	Wrapper	gum	ChewingGum
ribbon	String-Textile	candle	candle
cookie	Cookie	bowl	Bowl-Generic
candle	Candle	bottle	Bottle
marble	Marble-Ball	water	Water
stick	TreeBranch	straw	SiphonTube
apple	EdibleFruit	ring	RingShapedObject
flower	Flower-BotanicalPart	book	BookCopy
cup	DrinkingGlass	tube	Tube
hole	n/a	towel	Towel

8.9 APPENDIX I: GENERALIZATIONS CREATED FOR ENGLISH *IN* AND *ON* IN THE CROSS-LINGUISTIC EXPERIMENT WHEN NO TEST CASE IS EXCLUDED (INCLUDES ALL TRAINING CASES)

Best Generalization ON

Size: 4

--DEFINITE FACTS:

(rcc8-NTPP figure ground)

(Clingy figure)

--POSSIBLE FACTS:

--UNLIKELY FACTS:

Best Generalization ON

Size: 3

--DEFINITE FACTS:

(enclosesVertically ground figure)

(rcc8-EC figure ground)

--POSSIBLE FACTS:

67%: (leftOf figure ground)

33%: (enclosesHorizontally ground figure)

--UNLIKELY FACTS:

Best Generalization ON

Size: 3

--DEFINITE FACTS:

(enclosesVertically ground figure)

(definiteOverlapCase figure ground)

(rcc8-PO figure ground)

--POSSIBLE FACTS:

67%: (enclosesHorizontally ground figure)

33%: (rightOf figure ground)

--UNLIKELY FACTS:

Best Generalization ON

Size: 2

--DEFINITE FACTS:

(enclosesHorizontally ground figure)

(above figure ground)

(rcc8-EC figure ground)

--POSSIBLE FACTS:

--UNLIKELY FACTS:

Best Generalization ON

Size: 2

--DEFINITE FACTS:

(HollowCylindricalObject ground)
(leftOf figure ground)
(above figure ground)
(Covering-Object figure)
--POSSIBLE FACTS:
50%: (definiteOverlapCase figure ground)
50%: (rcc8-PO figure ground)
50%: (rcc8-EC figure ground)
--UNLIKELY FACTS:

Best Generalization ON
Size: 2
--DEFINITE FACTS:
(rcc8-EC figure ground)
(enclosesVertically ground figure)
(rightOf figure ground)
--POSSIBLE FACTS:
--UNLIKELY FACTS:

Best Generalization ON
Size: 2
--DEFINITE FACTS:
(rcc8-PO figure ground)
(enclosesHorizontally ground figure)
(below figure ground)
(definiteOverlapCase figure ground)
--POSSIBLE FACTS:
50%: (belowGrazingLine figure ground)
--UNLIKELY FACTS:

Best Generalization ON
Size: 2
--DEFINITE FACTS:
(below figure ground)
(rcc8-EC figure ground)
(belowGrazingLine figure ground)
--POSSIBLE FACTS:
50%: (Circling figure)
50%: (enclosesHorizontally ground figure)
50%: (enclosesHorizontally figure ground)
--UNLIKELY FACTS:

Best Generalization ON
Size: 2
--DEFINITE FACTS:
(rcc8-TPP figure ground)

(Circling figure)

--POSSIBLE FACTS:

--UNLIKELY FACTS:

Best Generalization ON

Size: 2

--DEFINITE FACTS:

(rcc8-PO figure ground)

(enclosesVertically ground figure)

(definiteOverlapCase figure ground)

--POSSIBLE FACTS:

50%: (rightOf figure ground)

50%: (leftOf figure ground)

--UNLIKELY FACTS:

Best Generalization IN

Size: 3

--DEFINITE FACTS:

(above figure ground)

(rcc8-EC figure ground)

--POSSIBLE FACTS:

67%: (enclosesHorizontally ground figure)

33%: (enclosesHorizontally figure ground)

33%: (RingShapedObject ground)

Best Generalization IN

Size: 2

--DEFINITE FACTS:

(rcc8-TPP figure ground)

--POSSIBLE FACTS:

50%: (Bowl-Generic Object-121)

50%: (Basin ground)

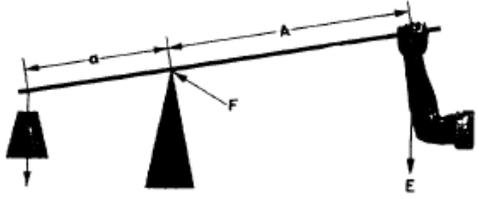
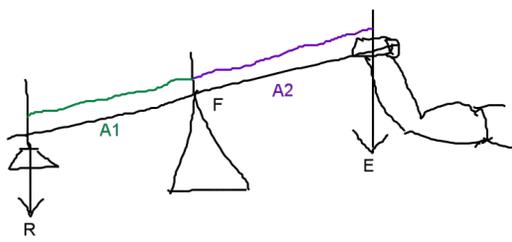
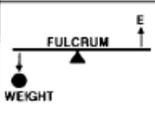
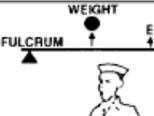
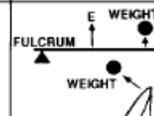
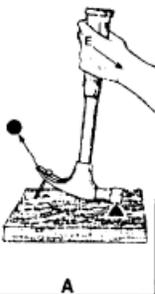
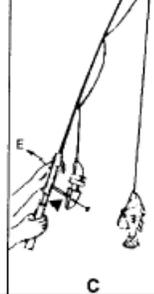
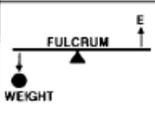
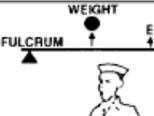
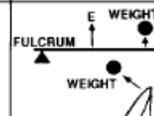
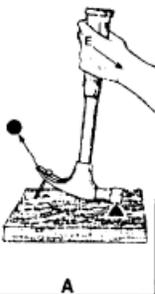
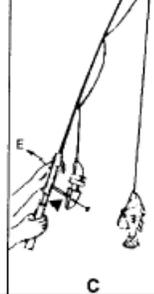
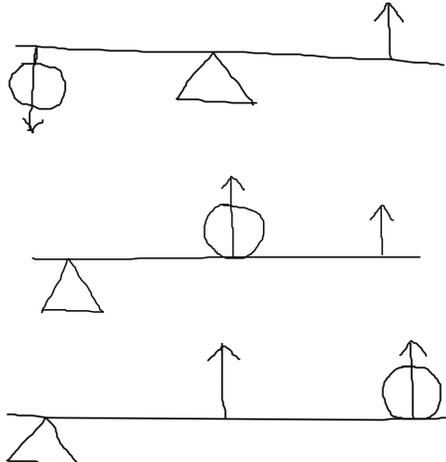
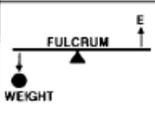
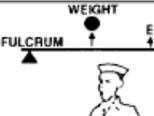
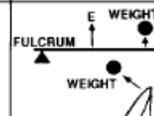
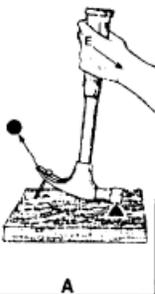
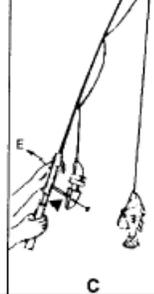
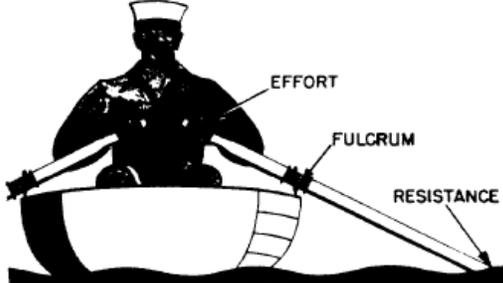
--UNLIKELY FACTS:

Exemplar <Exemplar IN>

Exemplar <Exemplar IN>

Exemplar <Exemplar IN>

8.10 APPENDIX J: DIAGRAMS FROM *BASIC MACHINES* CHAPTER 1

<p style="text-align: center;">Original Diagram</p>  <p style="text-align: center;">Figure 1-1.-A simple lever.</p>	<p style="text-align: center;">Sketch(es)*</p> 									
<table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th style="width: 33%;">FIRST CLASS LEVERS</th> <th style="width: 33%;">SECOND CLASS LEVERS</th> <th style="width: 33%;">THIRD CLASS LEVERS</th> </tr> </thead> <tbody> <tr> <td>  </td> <td>  </td> <td>  </td> </tr> <tr> <td>  <p>A</p> </td> <td>  <p>B</p> </td> <td>  <p>C</p> </td> </tr> </tbody> </table> <p style="text-align: center;">Figure 1-2.-Three classes of levers.</p>	FIRST CLASS LEVERS	SECOND CLASS LEVERS	THIRD CLASS LEVERS				 <p>A</p>	 <p>B</p>	 <p>C</p>	
FIRST CLASS LEVERS	SECOND CLASS LEVERS	THIRD CLASS LEVERS								
										
 <p>A</p>	 <p>B</p>	 <p>C</p>								
 <p style="text-align: center;">Figure 1-3.-Oars are levers.</p>										

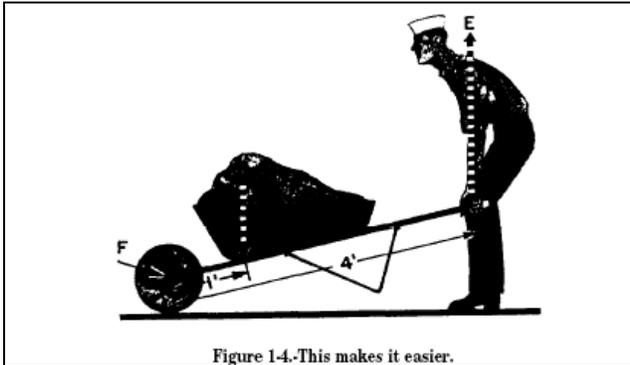


Figure 1-4.-This makes it easier.

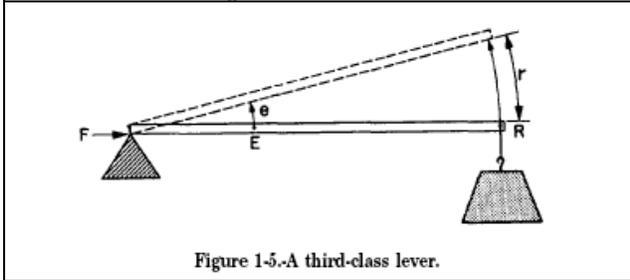
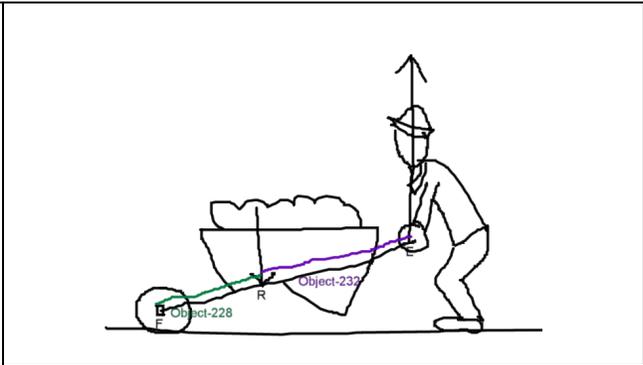


Figure 1-5.-A third-class lever.

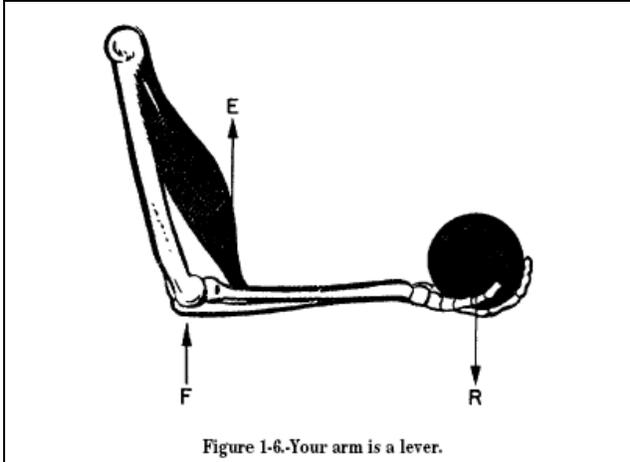
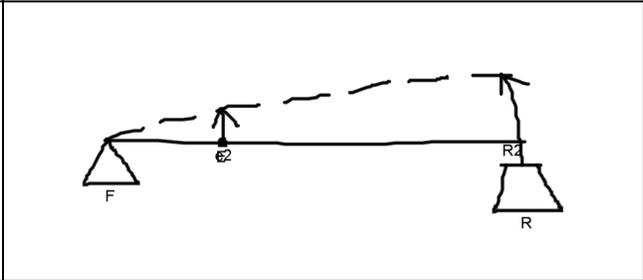


Figure 1-6.-Your arm is a lever.

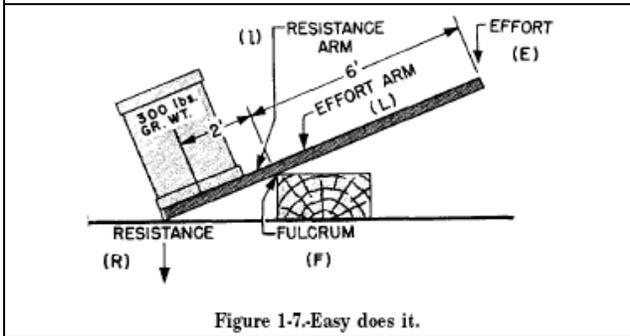
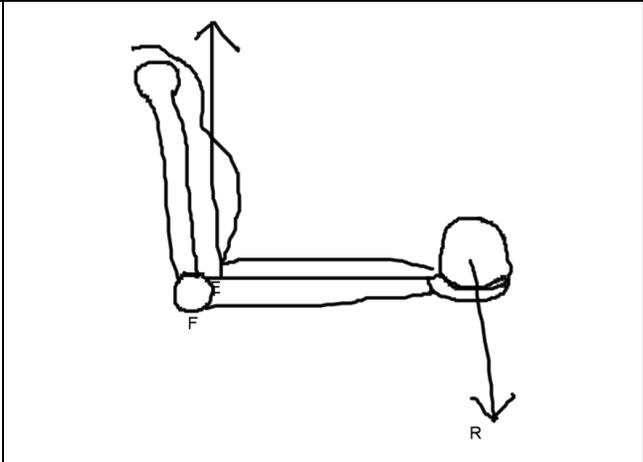
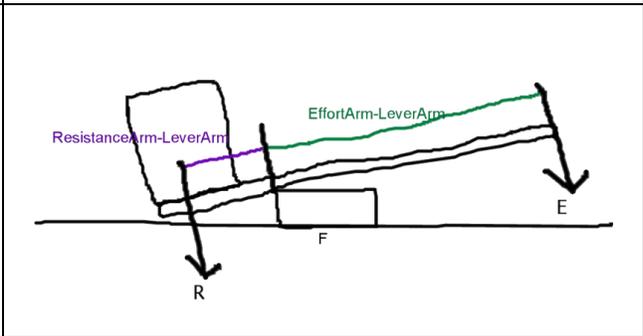


Figure 1-7.-Easy does it.



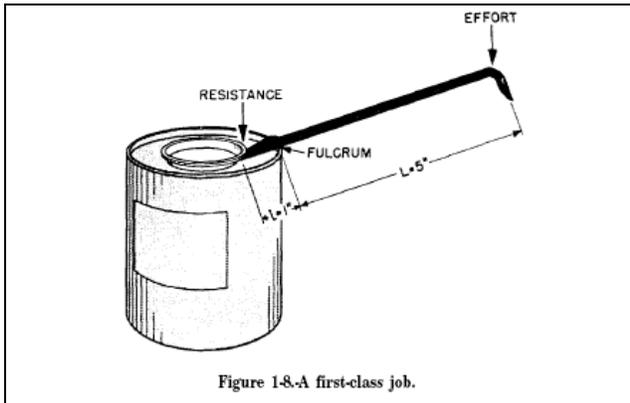


Figure 1-8-A first-class job.

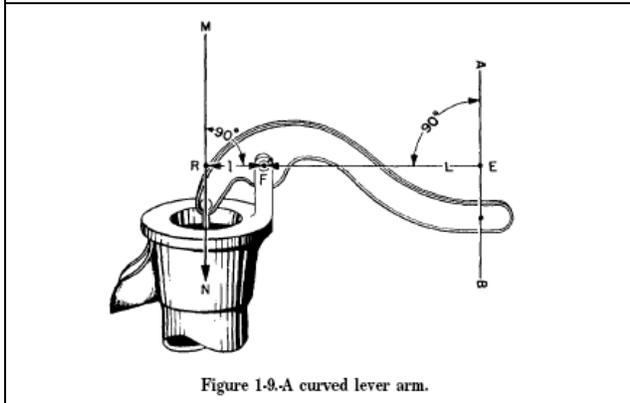
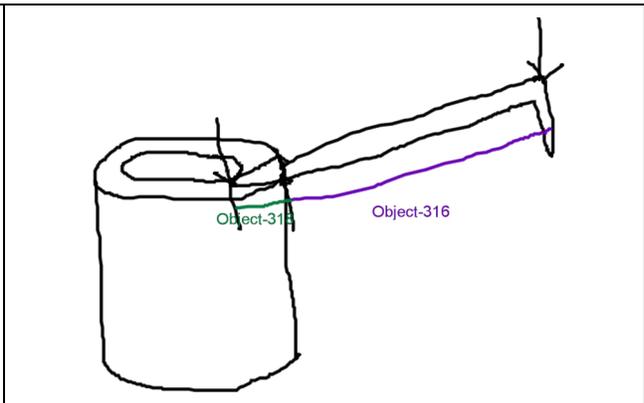


Figure 1-9-A curved lever arm.

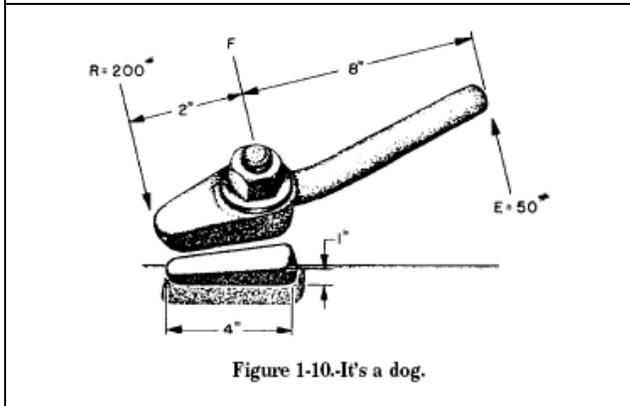
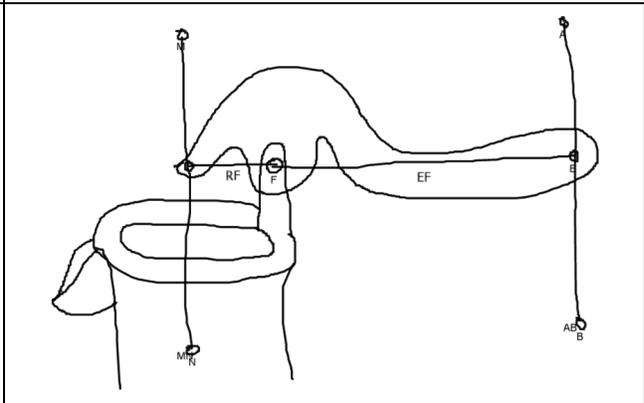


Figure 1-10-It's a dog.

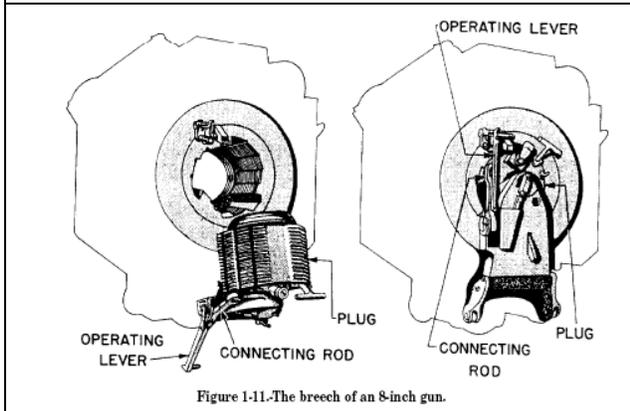
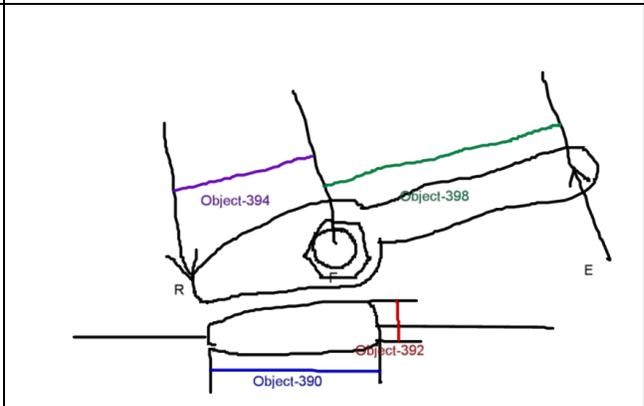


Figure 1-11-The breech of an 8-inch gun.



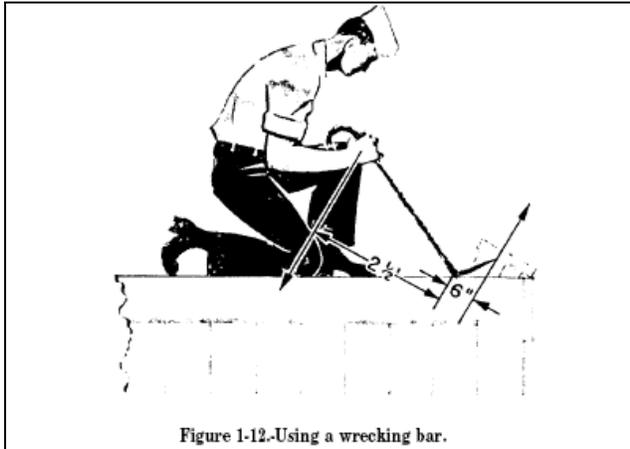


Figure 1-12.-Using a wrecking bar.

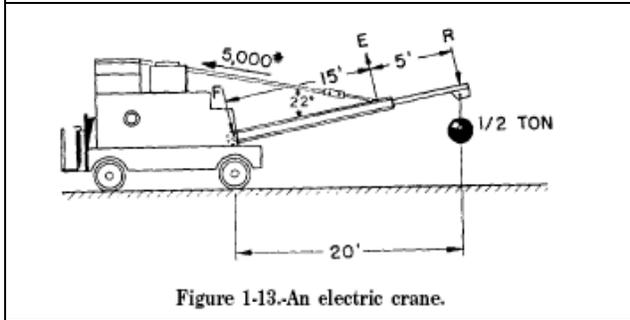
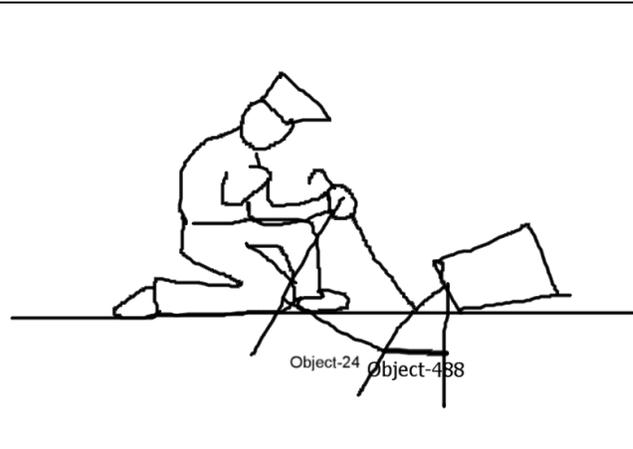


Figure 1-13.-An electric crane.

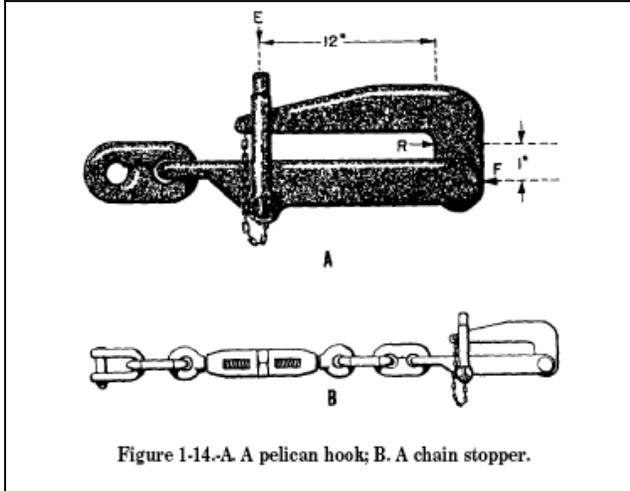
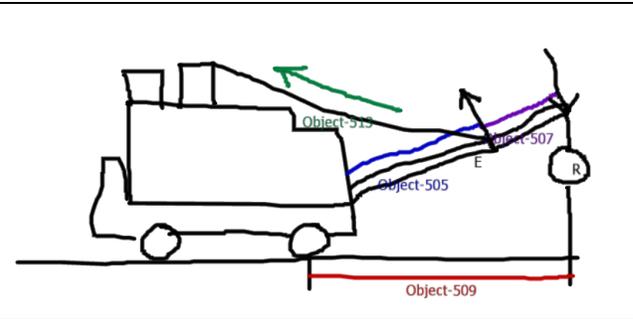
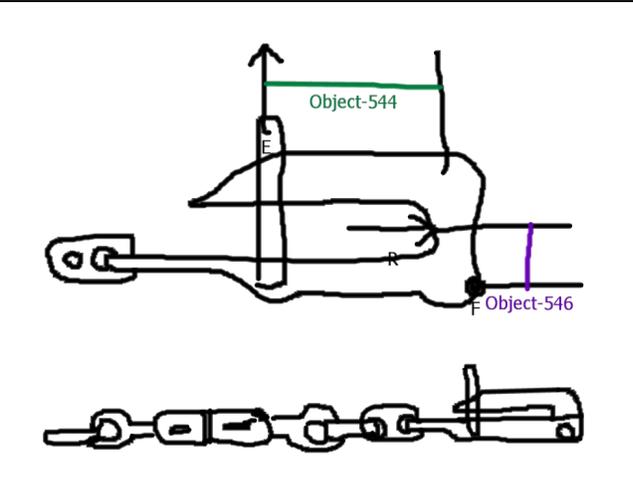


Figure 1-14.-A. A pelican hook; B. A chain stopper.



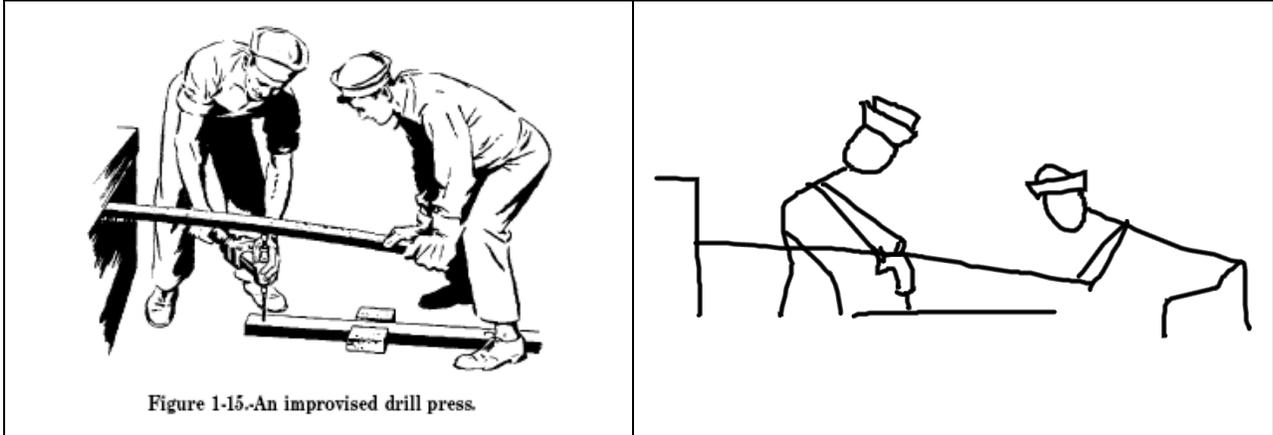


Figure 1-15.-An improvised drill press.

*In some cases, multi-part diagrams were broken into individual sketches for easier processing

8.11 APPENDIX K: HOMEWORK QUESTIONS FOR CHAPTER 1 IN *BASIC MACHINES*

(this is a subset of assignment 1 which covers chapters 1 through 4).

Correct answers are indicated in red

1-1. A chain hoist lifts a 300-pound load through a height of 10 feet because it enables you to lift the load by exerting less than 300 pounds of force over a distance of 10 feet or less.

1. True
2. **False**

1-2. When a chain hoist is used to multiply the force being exerted on a load, the chain is pulled at a faster rate than the load travels.

1. **True**
2. False

1-3. What are the six basic simple machines?

1. The lever, the block and tackle, the inclined plane, the engine, the wheel and axle, and the gear
2. The lever, the block and tackle, the wheel and axle, the screw, the gear, and the eccentric
3. **The lever, the block and tackle, the wheel and axle, the inclined plane, the screw, and the gear**
4. The lever, the inclined plane, the gear, the screw, the fulcrum, and the torque

1-4. Which of the following basic principles is recognized by physicists as governing each simple machine?

1. The wedge or the screw
2. The wheel and axle or the gear
3. **The lever or the inclined plane**
4. The block and tackle or the wheel and axle

1-5. Which of the following simple machines works on the same principle as the inclined plane?

1. **Screw**
2. Gear
3. Wheel and axle
4. Block and tackle

1-6. The fundamentally important points in any lever problem are (1) the point at which the force is applied, (2) the fulcrum, and (3) the point at which the:

1. lever will balance
2. resistance arm equals the effort arm
3. mechanical advantage begins to increase
4. **resistance is applied**

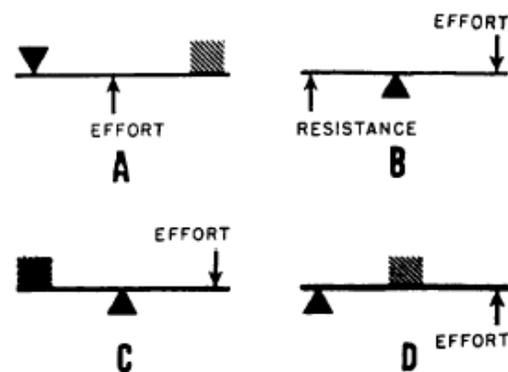


Figure 1A.

QUESTIONS 1-7 THROUGH 1-9 RELATE TO THE DRAWINGS IN FIGURE 1A.

1-7. Which, if any, of the following parts illustrates a first class lever?

1. A
2. **B or C**
3. D
4. None of the above

1-8. Which part illustrates a Second-class lever?

1. **D**
2. C
3. B
4. A

1-9. What part illustrates a third-class lever?

1. **A**
2. B
3. C
4. D

1-10. Which of the following classes of levers should you use to lift a large weight by exerting the least effort?

1. First-class
2. Second-class
3. **First- or second-class**
4. Third-class

1-11. You will find it advantageous to use a third-class lever when the desired result is

1. a transformation of energy
2. **an increase in speed**
3. a decrease in applied effort
4. a decrease in speed and an increase in applied effort

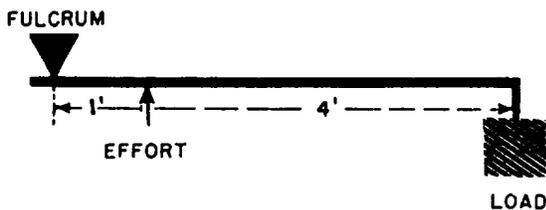


Figure 1B

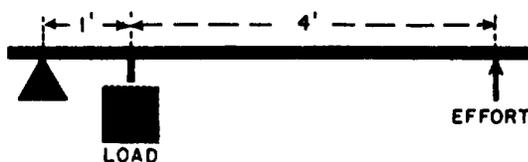


Figure 1C

IN ANSWERING QUESTIONS 1-12 THROUGH 1-14, SELECT THE CORRECT ARM MEASUREMENTS FROM FIGURES 1B AND 1C.

1-12. Effort arm in figure 1B

1. **1 ft**
2. 3 ft
3. 4 ft
4. 5 ft

1-13. Resistance arm in figure 1B

1. 1 ft
2. **3 ft**

3. 4 ft
4. **5 ft**

1-14. Resistance arm in figure 1C

1. **1 ft**
2. 3 ft
3. 4 ft
4. 5 ft

1-15. Two boys find that they can balance each other on a plank if one sits six feet from the fulcrum and the other eight feet. The heavier boy weighs 120 pounds. How much does the lighter boy weigh?

1. **90 lb**
2. 106 lb
3. 112 lb
4. 114 lb

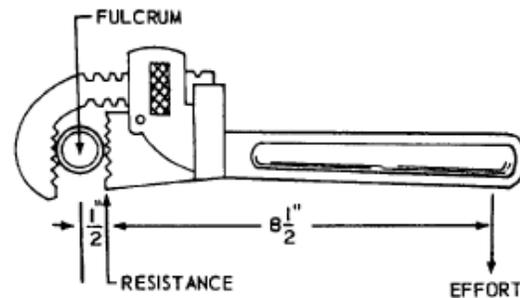


Figure 1D

1-16. With the aid of the pipe wrench shown in figure 1D, how many pounds of effort will you need to exert to overcome a resistance of 900 pounds?

1. 25 lb
2. **50 lb**
3. 75 lb
4. 100 lb

Questions 1-17 and 1-18 are related to a 300-pound load of firebrick stacked on a wheelbarrow. Assume that the weight of the firebrick is centered at a point and the barrow axle is 1 1/2 feet forward of the point.

1-17. If a Seaman grips the barrow handles at a distance of three feet from the point, how many total pounds will the Seaman have to lift to move the barrow?

1. 65 lb
2. **100 lb**

3. 150 lb
4. 300 lb

1-18. If a Seaman grasps the handles 3 1/2 feet from the point where the weight is centered, how many pounds of effort will be exerted?

1. 50 lb
2. 90 lb
3. 100 lb
4. 120 lb

1-19. In lever problems, the length of the effort arm multiplied by the effort is equal to the length of the

1. resistance arm multiplied by the effort
2. resistance arm multiplied by the resistance
3. effort arm multiplied by the resistance arm
4. effort arm multiplied by the Resistance

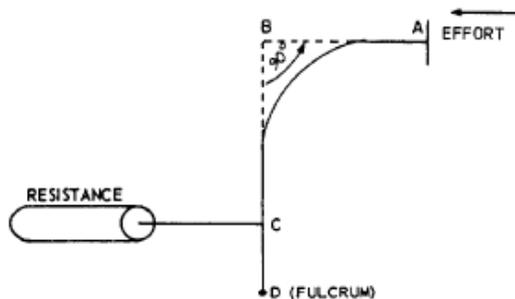


Figure 1E.—A curved lever.

1-20. The length of the effort arm in figure 1E is equal to the length of the

1. curved line from A to C
2. curved line from A to D
3. straight line from B to C
4. straight line from B to D

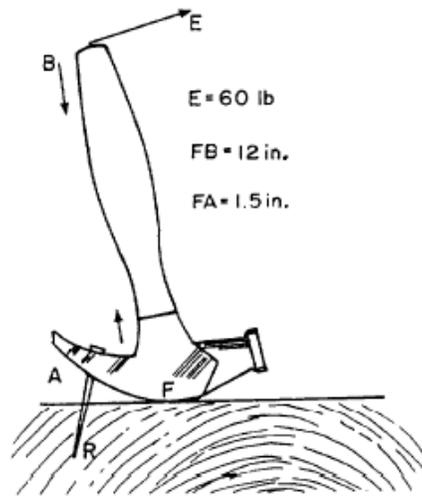


Figure 1F

1-21. Refer to figure 1F. If a person exerts at point B a pull of 60 pounds on the claw hammer shown, what is the resistance that the nail offers?

1. 60 lb
2. 120 lb
3. 480 lb
4. 730 lb

1-22. Which of the following definitions describes the mechanical advantage of the lever?

1. Effort that must be applied to overcome the resistance of an object divided by the resistance of the object
2. Amount of work obtained from the effort applied
3. Gain in power obtained by the use of the lever
4. Resistance offered by an object divided by the effort which must be applied to overcome this resistance

1-23. The mechanical advantage of levers can be determined by dividing the length of the effort arm by the

1. distance between the load and the point where effort is applied
2. distance between the fulcrum and the point where effort is applied
3. distance between the load and the fulcrum
4. amount of resistance offered by

the object

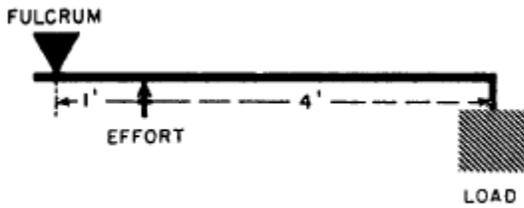


Figure 1G

1-24. The mechanical advantage of the lever in figure 1G is

1. one-fifth
2. one-fourth
3. four
4. five

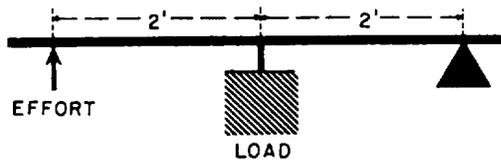


Figure 1H

1-25. The mechanical advantage of the lever in figure 1H is

1. one
2. two
3. one-half
4. one-fourth

1-26. The mechanical advantage of the lever pictured in figure 1J is

1. five
2. six
3. seven
4. one-sixth

1-27. The combination dog and wedge of textbook figure 1-10 is a complex machine since it consists of which two simple machines?

1. Lever and the screw
2. Two first-class levers
3. Lever and the inclined plane
4. One first-class lever and one second-class lever

Information for questions 1-28 and 1-29: The handle of a hatch dog is 9 inches long. The short arm is 3 inches long.

1-28. What is the mechanical advantage of

the hatch dog?

1. 12
2. 27
3. 3
4. 9

1-29. With how much force must you push down on the handle to exert 210 pounds force on the end of the short arm?

1. 105 lb
2. 80 lb
3. 70 lb
4. 25 lb

8.12 APPENDIX L: KNOWLEDGE ADDED TO THE KNOWLEDGE BASE TO FACILITATE KNOWLEDGE CAPTURE

(in-package :data)

(in-microtheory EANLU :exclude-globals t)

(isa BasicMachine Collection)

(gens BasicMachine MechanicalDevice)

(isa InclinedPlane Collection)

(isa Wedge-Tool Collection)

(gens InclinedPlane MechanicalDevice)

(gens Wedge-Tool MechanicalDevice)

(isa HatchDog Collection)

(gens HatchDog MechanicalDevice)

(isa Lever-FirstClass Collection)

(isa Lever-SecondClass Collection)

(isa Lever-ThirdClass Collection)

(gens Lever-FirstClass Lever)

(gens Lever-SecondClass Lever)

(gens Lever-ThirdClass Lever)

(isa FirstClass Individual)

(isa FirstClass NonNumericQuantity)

(isa FirstClass EvaluativeQuantity)

(isa SecondClass Individual)

(isa SecondClass NonNumericQuantity)

(isa SecondClass EvaluativeQuantity)

(isa ThirdClass Individual)

(isa ThirdClass NonNumericQuantity)

(isa ThirdClass EvaluativeQuantity)

(isa LeverArm Collection)

(isa EffortArm-LeverArm Collection)

(isa ResistanceArm-LeverArm Collection)

(gens EffortArm-LeverArm LeverArm)

(gens ResistanceArm-LeverArm LeverArm)

(isa MechanicalAdvantage Individual)

;;;;;;;;;;;;;

;;; nouns

(isa Seesaw-TheWord LexicalWord)

(isa Seesaw-TheWord EnglishWord)

(posForms Seesaw-TheWord CountNoun)

(denotation Seesaw-TheWord CountNoun 85 PlaygroundEquipment)

(denotation Pivot-TheWord CountNoun 85 Fulcrum)

(isa Sailor-TheWord LexicalWord)

(isa Sailor-TheWord EnglishWord)

(posForms Sailor-TheWord CountNoun)

(denotation Sailor-TheWord CountNoun 85 CrewMemberOnShip)

(multiWordString (TheList "inclined") Plane-TheWord CountNoun InclinedPlane)

(multiWordString (TheList "mechanical") Advantage-TheWord CountNoun MechanicalAdvantage)

(multiWordString (TheList "lever") Arm-TheWord CountNoun LeverArm)

(multiWordString (TheList "effort") Arm-TheWord CountNoun EffortArm-LeverArm)

(multiWordString (TheList "resistance") Arm-TheWord CountNoun ResistanceArm-LeverArm)

(multiWordString (TheList "hatch") Dog-TheWord CountNoun HatchDog)

(denotation Arm-TheWord CountNoun 0 LeverArm)

(denotation Wedge-TheWord CountNoun 0 Wedge-Tool)

(denotation Resistance-TheWord MassNoun 85 Weight)

(isa Oarlock-TheWord LexicalWord)

(isa Oarlock-TheWord EnglishWord)

(posForms Oarlock-TheWord CountNoun)

(denotation Oarlock-TheWord CountNoun 85 Joint-Junction)

(multiWordString (TheList "wheel" "and") Axle-TheWord CountNoun WheelAndAxle)

(compoundString Pair-TheWord (TheList "of" "pliers") CountNoun Pliers)

(verbSemTrans Start-TheWord 85 (PPCompFrameFn TransitivePPFrameType At-TheWord)
(startOfPath :OBLIQUE-OBJECT :SUBJECT))

(denotation Load-TheWord Noun 0 Weight)

.....
;;; verbs

(verbSemTrans Move-TheWord 85 IntransitiveVerbFrame
(and (isa :ACTION Movement-TranslationEvent)
(distanceTranslated :ACTION :OBJECT)
(primaryObjectMoving :ACTION :SUBJECT)))

(verbSemTrans Move-TheWord 0 IntransitiveVerbFrame
(and
(isa :ACTION Movement-TranslationEvent)
(primaryObjectMoving :ACTION :SUBJECT)
(distanceTranslated :ACTION :MEASURE)))

```
(verbSemTrans Equal-TheWord 85 IntransitiveVerbFrame
  (equals :SUBJECT :ACTION))
```

```
(adjSemTrans Equal-TheWord 85 RegularAdjFrame
  (equals :SUBJECT :ACTION))
```

```
(verbSemTrans Divide-TheWord 85 (PPCompFrameFn TransitivePPFrameType By-TheWord)
  (QuotientFn :SUBJECT :NOUN))
```

```
(isa DivisionEvent Event)
(isa dividend ActorSlot)
(isa divisor ActorSlot)
(arg1Isa dividend DivisionEvent)
(arg1Isa divisor DivisionEvent)
```

```
(verbSemTrans Divide-TheWord 86 IntransitiveVerbFrame
  (and
    (isa :ACTION DivisionEvent)
    (dividend :ACTION :SUBJECT)
    (divisor :ACTION :OBJECT)))
```

```
(verbSemTrans Curve-TheWord 0 RegularVerbFrame
  (physicalStructuralFeatures :NOUN Curved))
```

```
.....
;;; adjectives
```

```
(denotation First-TheWord OrdinalAdjective 1 FirstClass)
(denotation First-Class-TheWord Adjective 0 FirstClass)
(denotation Second-Class-TheWord Adjective 0 SecondClass)
(denotation Third-Class-TheWord Adjective 0 ThirdClass)
```

```
;;; want to do the more general case
```

```
(multiWordString (TheList "first") Class-TheWord CountNoun FirstClass)
(multiWordString (TheList "first" "class") Lever-TheWord CountNoun Lever-FirstClass)
```

```
(multiWordString (TheList "second") Class-TheWord Adjective SecondClass)
(multiWordString (TheList "second" "class") Lever-TheWord CountNoun Lever-SecondClass)
```

```
(multiWordString (TheList "third") Class-TheWord Adjective ThirdClass)
(multiWordString (TheList "third" "class") Lever-TheWord CountNoun Lever-ThirdClass)
```

```
(denotation Light-TheWord Adjective 85 (LowAmountOfFn Weight))
(denotation Basic-TheWord Adjective 85 ConfigurationTypeByComplexity-NoAmount)
```

(denotation Complex-TheWord Adjective 85 ConfigurationTypeByComplexity-HighAmount)

(adjSemTrans Perpendicular-TheWord 85 RegularAdjFrame
 (perpendicularObjects :NOUN :OBLIQUE-OBJECT))

;;;;;;;;;;;;
 ;;; prepositions

(denotation Through-TheWord Preposition 85 trajectoryPassesThrough)

(prepSemTrans Through-TheWord 85 Post-NounPhraseModifyingFrame
 (trajectoryPassesThrough :NOUN :OBJECT))

(prepSemTrans From-TheWord 1 Post-NounPhraseModifyingFrame
 (thereExists :THIS
 (and
 (isa :THIS Distance)
 (distanceBetween :NOUN :OBJECT :THIS))))

(arity distanceBetween 3)

;;;;;;;;;;;;
 ;;; adverbs

(definitionInDictionary COMLEX31Lexicon "farther" (farther (adverb (orth "farther") (root far2) (modif
 clausal-adv) (comparative +) (gradable +) (manner-adv +)) farther2))

(denotation Less-TheWord Adverb 0 lessThan)

(adverbSemTrans Less-TheWord 0 RegularAdjFrame
 (denotesRelation-Underspecified :SUBJECT lessThan))

(denotation Fast-TheWord Adverb 85 (HighAmountOfFn Speed))

(denotation Further-TheWord Adverb 85 (HighAmountOfFn Distance))

;;;;;;;;;;;;
 ;;; test setup predicates

(arity sketchForDiscourse 2)
 (arity discourseForChapter 2)
 (isa LeverEffortProblem Collection)
 (isa LeverMAPProblem Collection)

;;;;;;;;;;;;
 ;;; knowledge needed for test questions

(arity sketchForQuery 2)
(arity multipleChoiceCorrectAnswer 2)
(arity determineMeasurementFromSketch 2)
(arity previousDRSInChapter 3)
(arity mechanicalAdvantageOf 1)
(arity workedSolutionMtForTestMt 2)
(arity workedSolutionForKBContentTest 2)
(arity equationForSolution 2)
(arity mathEquals 2)

(arity comparee 2)
(arity comparer 2)

8.13 APPENDIX M: FILTER USED TO EXTRACT BOOKKEEPING INFORMATION FROM SKETCH CASES

```
(defun remove-mmckap-filter (reasoner fact)
  (declare (ignore reasoner))
  (when (and (consp fact)
             (eq (car fact) 'd::ist-Information)
             (consp (third fact)))
    ;; only case facts are useful here
    (let ((inner-fact (third fact)))
      (and (not (filter-mmckap-info? inner-fact))
           (not (filter-isa? inner-fact))))))

(defparameter *initial-mmckap-filter*
  'd::{askConceptualForBinaryVisualRelation
    bboxLastModifiedTime
    connectedGlyphGroupMember
    connectedGlyphGroupTangentialConnection
    containedGlyphGroupContainer
    containedGlyphGroupInsider
    containedGlyphGroupTangentialInsider
    defaultUnits
    directionalSignature
    entityTypeLastModifiedTime
    glyphAssociation
    glyphGraphCWA
    glyphGraphEdgesFor
    glyphRepresentsObject
    hasRCC8Relation
    inkLastModifiedTime
    kbDateModified
    nameStringLastModifiedTime
    needVisualPositionalRelation
    nuSketchCaseID
    nuSketchCaseNotes
    nuSketchCreationMachine
    nuSketchCreator
    nuSketchLayerOf
    nuSketchLayerForCase
    nuSketchSketchOf
    nuSketchSketchForCase
    sketchCreatedWithVersion
    sketchFor
    sketchModifiedWithVersion
    sketchRepresentsObject
```

```

subCaseOf
subSketchFor
subSketchGroupFor
subSketchHasPose
subSketchHasGenre
subSketchRepresentsObject
q-2D-orientation
q-roundness
userCWA
voronoiFor))

```

```
(defparameter *third-isa-atom-filter*
```

```

'd::(Case
  ConnectedGlyphGroup
  ContainedGlyphGroup
  ContainedGlyphGroupFn
  ConnectedGlyphGroupFn
  Glyph
  Individual
  Latitude
  Longitude
  LookingFromSide-SubSketch
  NotVeryRoundGlyph
  NotVeryRound
  NuSketchBundle
  NuSketchCase
  NuSketchGlyph
  NuSketchLayer
  NuSketchSketch
  PhysicalView-SubSketch
  SomewhatRoundGlyph
  SomewhatRound
  Sketch-Drawing
  StaticSituation
  SubSketch
  SubSketchGroup
  VeryRoundGlyph
  VeryRound
  QDiagonalDownwardGlyph
  QDiagonalUpwardGlyph
  QHorizontalGlyph
  QVerticalGlyph
  SmallSizeGlyph
  LargeSizeGlyph
  Yard-UnitOfMeasure))

```

```
(defun filter-mmkcip-info? (fact)
  (member (car fact) *initial-mmkcip-filter* :test #'eq))

(defun filter-isa? (fact)
  (and (eq (car fact) 'd::isa)
       (member (third fact) *third-isa-atom-filter* :test #'eq)))
```