Spring 2007

Diagnosing Reading strategies: Paraphrase Recognition

Chutima Boonthum
Old Dominion University

DIAGNOSING READING STRATEGIES:

PARAPHRASE RECOGNITION

by

Chutima Boonthum
B.S. March 1997, Srinakharinwirot University
M.S. May 2000, Illinois State University

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirement for the Degree of

DOCTOR OF PHILOSOPHY

COMPUTER SCIENCE

OLD DOMINION UNIVERSITY
May 2007

Approved by:

Irwin B. Levinstein (Director)

Shunichi Toida (Member)

Stewart N.T. Shen (Member)

Danielle S. McNamara (Member)

Johan Bollen (Member)

UMI Number: 3264822

ABSTRACT

DIAGNOSING READING STRATEGIES: PARAPHRASE RECOGNITION

Chutima Boonthum
Old Dominion University, 2007
Director: Dr. Irwin B. Levinstein

Paraphrase recognition is a form of natural language processing used in tutoring, question answering, and information retrieval systems. The context of the present work is an automated reading strategy trainer called iSTART (Interactive Strategy Trainer for Active Reading and Thinking). The ability to recognize the use of paraphrase – a complete, partial, or inaccurate paraphrase; with or without extra information – in the student's input is essential if the trainer is to give appropriate feedback. I analyzed the most common patterns of paraphrase and developed a means of representing the semantic structure of sentences. Paraphrases are recognized by transforming sentences into this representation and comparing them. To construct a precise semantic representation, it is important to understand the meaning of prepositions. Adding preposition disambiguation to the original system improved its accuracy by 20%. The preposition sense disambiguation module itself achieves about 80% accuracy for the top 10 most frequently used prepositions.

The main contributions of this work to the research community are the preposition classification and generalized preposition disambiguation processes, which are integrated into the paraphrase recognition system and are shown to be quite effective. The recognition model also forms a significant part of this contribution. The present effort includes the modeling of the paraphrase recognition process, featuring the Syntactic-Semantic Graph as a sentence representation, the implementation of a significant portion of this design demonstrating its effectiveness, the modeling of an effective preposition classification based on prepositional usage, the design of the generalized preposition disambiguation module, and the integration of the preposition disambiguation module into the paraphrase recognition system so as to gain significant improvement.

This dissertation is dedicated to my parents,
Koon & Kanokwan Boonthum.

ACKNOWLEDGMENTS

I would like to thank Dr. Shunichi Toida and Dr. Irwin B. Levinstein for their guidance and research support since January 2003. Due to the absence of Computational Linguistics in our department, it has been difficult to overcome the challenges that have arisen while completing this work. Nevertheless, both of them have been very supportive and have shown me how to become a good researcher and develop an inquisitive mind. I have learned that regardless of how much we think we know, there is always room to learn. This has been a long journey, but a very rewarding experience. There have been a lot of discussions, agreements and arguments; my advisors and I have learned a lot. A special thanks to Dr. Levinstein for his support and in funding trips to present this dissertation work.

I also would like to thank Dr. Danielle S. McNamara. Her iSTART project, funded by NSF, has motivated me to pursue this area of research, specifically the ability to recognize a paraphrase used by the trainee in the system. Had there been no iSTART, I would have been forced to choose a different topic for my dissertation. Also, Dr. McNamara has been a role model to me, that women can be as diligent, intelligent, and successful in conducting quality research as men, proven by her multi-million dollar funding from NSF and IES, and over a hundred publications.

I also would like to thank Mrs. Janet Brunelle on her support. She and Dr. Larry Wilson were my first two supervisors in the department. Mrs. Brunelle has welcomed me to be part of her professional life (as her advising assistant) as well as her personal life (as her family friend). She has shown me by example how to be an extraordinary people person. When dealing with students' problems, she always manages to find the best way to solve them. And, when her colleagues have issues, she is an outstanding moderator.

Lastly, I would like to thank to the iSTART team both at ODU and the University of Memphis, friends, and family, who have always been there for me. Their support has been valuable to me.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

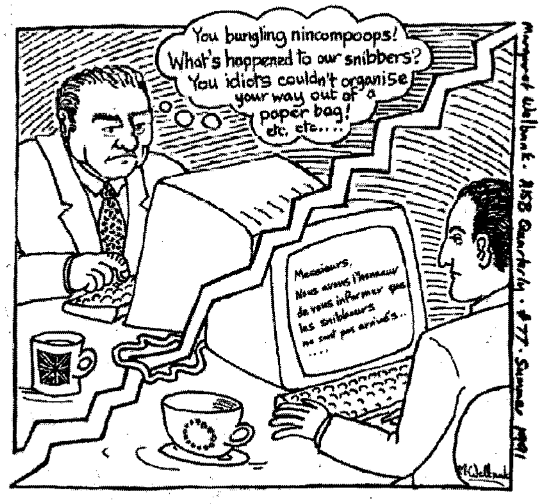
LIST OF FIGURES

Figure Page

CHAPTER 1

INTRODUCTION

When two expressions describe the same situation, each is a paraphrase of the other. Paraphrasing is a common linguistic mechanism used to minimize the language barrier, for example when translating between languages, as shown in Figure 1, and is frequently used for referring to other people's work or statements.



Bridging the Language Barrier with Intelligent Systems

Figure 1:[1] Language Translation.

So, what is a paraphrase? The answer starts with "*a paraphrase is a restatement or a way to talk about the same situation in a different way*" although "the same situation" and "a different way" can be interpreted in different ways (Hurst, 2003). Academic writing centers (ASU Writing Center, 2000; Quality Writing Center, 2002; BAC Writing Center, 2002; USCA Writing Room, 2002; Hawes, 2003) provide a number

---

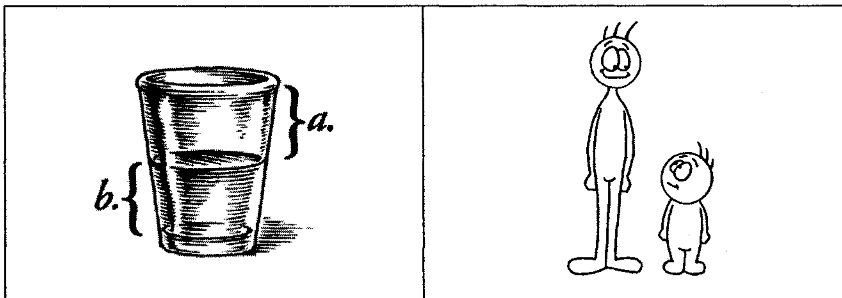
[1] This image is retrieved from Hurst (2003).

The journal model for this dissertation is the Journal of Artificial Intelligence Research.

of paraphrase characterizations, such as using synonyms, changing part-of-speech, reordering ideas, breaking a sentence into smaller ones (which includes combining sentences into one), using definition, or even using an example. The characterization common to almost all of them is that "*paraphrasing means restating ideas in our own words.*" This can be achieved by exchanging the original words with one's own words. The writer can use synonyms or different word forms or change the sentence structure to create your own rhythm. Out of the writing centers mentioned above, Hawes (2003) is perhaps the only source states that a brief definition or an example is a part of paraphrasing. According to McNamara (2004), using definitions or examples which include knowledge outside the text is considered to be an *elaboration* rather than a paraphrase. Stede (1996) says "*if two utterances are paraphrases of one another, they have the same content and differ only in aspects that are somehow secondary.*" A similar question can be asked on how to interpret "the same content" and "secondary aspects."

The Problems of Paraphrase Recognition

Why is it difficult to develop the paraphrase recognition system, which can be applied in any applications? First, it is because the definition of "paraphrase" is not precise and each definition is mostly tied to an application. In question answering systems, a student's answer is compared with an expected answer. An exact match is preferable, but a paraphrase is credited as well. In tutoring systems, a student's input is compared to an ideal response. During this comparison, it is rare for an exact match and the system is required to give an appropriate and accurate response, so a paraphrase is preferable. Hence, if the student's input is a paraphrase of the ideal response, it indicates that the student has the same idea along the line of what the system is expected from them. It is obvious that both applications require different set of paraphrase definitions: question answering systems may have stricter definitions while tutoring systems have looser ones. Second, the coverage required in each application is different. From previous examples, the question answer systems would require complete coverage of the student's answer to the expected answer whereas tutoring systems may require only partial coverage, focusing on coverage of key information. Third, the definition of synonyms plays a part in the paraphrase recognition. On the one hand, some synonyms are interchangeable, that is, they can be used without changing any meaning of a sentence. On the other hand, some synonyms, perhaps better described as *near-synonyms*, change the meaning of a sentence. Fourth, different aspects used to describe the same situation (called *canonical paraphrases*) also increase difficulty to some extent in recognizing a paraphrase, as shown in Figure 2: (a) fullness versus emptiness and (b) tall versus short.



(a)[2] "The glass is half-empty." vs. "The glass is half-full."

(b)[3] "Tom is taller than Tim." vs. "Tim is shorter than Tom."

Figure 2: Canonical Paraphrases.

Why must a paraphrase be recognized? One answer is to automate essay grading or replace a human-led trainer with an automated trainer. In particular with a reading strategy trainer, paraphrase recognition will improve the feedback and properly guide the trainees throughout the curriculum. Instead of giving a general (and largely meaningless) response, such as "Ok", "That's fine", "That's good", more specific feedback can be provided, such as "That's a good paraphrase" or "You are missing some information." In question answering systems, recognizing a paraphrase is a way to score the student's answer against the ideal answer: scoring information content rather than grammatical form. Once the paraphrase recognition module is in place, the scoring process can be done automatically rather than having it manually graded by experts.

[2] This image is retrieved from http://www.penart.com/a2z_stockfiles/g_folder/glasshalffull.gif
[3] This image is retrieved from http://tell.fll.purdue.edu/JapanProj/FLClipart/Adjectives/tall&short.gif

Why is recognizing a paraphrase correctly important? One motivation is to be able to provide accurate scoring in question answer systems and appropriate guidance in tutoring systems. One question that can be raised: what do these tutoring systems mean by being correct? A simple answer is that the student's input should cover the ideal answer (or ideal response) as much as possible. Simple word matching may work in the case of a short-answer question, while complex word matching (including co-occurrences, word order, stemming, and spelling) may be needed for long-answers or essay questions. For essay questions, deeper semantics for answers may be required. In a reading strategy training program (such as iSTART) that teaches various strategies including paraphrasing, the ability to recognize a correct paraphrase as well as an attempt paraphrase is essential to the feedback system. If the system responds incorrectly and/or misguides the student through the curriculum, the student could learn the wrong thing and ultimately receive no gain from the system. Therefore, recognizing a paraphrase correctly should improve the feedback system. In question answering systems, recognizing paraphrase correctly will provide the students real-time feedback while they are taking the tests and move the assessment tool from a proactive to an active one, leading to an automated grading system.

Motivation

This work on paraphrase recognition is inspired by the phase of the iSTART project (Interactive Strategy Trainer for Active Reading and Thinking, described below) in which the student practices producing (*i.e.*, typing) explanations. The system evaluates the student's explanation: it understands the student's input and gives appropriate feedback. Other applications, such as question answering and information retrieval can also use paraphrase recognition as described in a section below.

iSTART is a web-based automated reading strategy trainer. It follows the SERT (Self-Explanation Reading Training) methodology developed by McNamara (2004) as a way to improve high school students' reading ability by teaching them to use active reading strategies (comprehension monitoring, paraphrasing, bridging, elaboration, and prediction) in explaining difficult texts.

In both human-led and iSTART SERT training, the student is given an introduction to these reading strategies followed by a demonstration of how these strategies can be used in reading science texts. After that, the student has an opportunity to practice the strategies by reading a given text and explaining it sentence by sentence while receiving some guidance from a trainer. The existing evaluation system uses word-matching and Latent Semantic Analysis (LSA) to evaluate the students' responses. The results from previous iSTART experiments show that the evaluation system could be improved. There were cases where the explanations were good according to a human evaluator but were rejected by the computerized trainer for being too short, irrelevant, or too similar to the original (or the given) sentence. For example, for a sentence "Coal is the most abundant of the fossil fuels" and a student's explanation is "it was very important in the survival of people back many years." The computerized evaluation rejected this explanation as being irrelevant while the human evaluator gave a "good" rate (a score of 2, detailed explanation of scores is in Chapter 8). Contrariwise, there were some cases when the explanations were poor but given a high rating by the trainer. With the same given sentence and a student's explanation "A good way to start, some background knowledge on coal," human evaluator rated the quality of explanation as being an "ok" (a score of 1), while the computerized gave a "good" rate (a score of 2). These misjudgments occur because the computerized trainer does not truly understand the explanation because its methods of analysis completely ignore the sentence structure. With deeper understanding of the input, the trainer would be able to handle both problems of the students' explanations.

Depending on the level of the student (as determined from pretest scores or performance in the earlier modules), the trainer will use the results of this proposed paraphrasing evaluation in different ways. A student with a poor background (*e.g.*, low level reading skills, little prior knowledge) may be praised for using a moderately successful paraphrase while a more advanced student would be encouraged to do more. Although the SERT methodology does not consider a paraphrase by itself to be an explanation, being able to paraphrase is considered a great achievement for the students who have no experience with any of these reading strategies. Therefore, it is necessary for the iSTART development team to be able to recognize the use of paraphrases in the student's explanation.

Objectives

The main goal of this research is to be able to recognize different types of paraphrase. As shown in Figure 3, there are two main tasks involved in the recognition process: (1) constructing internal representations of the target sentence and student's explanation and (2) recognizing various paraphrasing patterns.

Constructing an Internal Representation. To construct an internal representation, the natural language is transformed into another knowledge representation with which we can analyze and perform logical reasoning during the recognition process. This construction process involves two steps: (1) parsing the given input (with a sentence parser) and (2) generating a knowledge representation for this input (using a representation generator). The *Sentence Parser* will analyze an input and return an output with syntax tags and morphological tags. The output will then be transformed into an appropriate knowledge representation. The *Representation Generator* will be implemented according to the chosen knowledge representation.

Recognizing A Paraphrase. There are a number of common patterns of paraphrasing (details are in Chapter 2), such as using synonyms and changing voice (active vs. passive), and it is important that the system is able to recognize the use of one or more of these patterns in comparing two sentences. To recognize the usage of each paraphrase, a set of paraphrase patterns have been defined for this research, along with a recognition model for these paraphrase patterns. The input to this process is a pair of outputs from the Representation Generator. The recognition process involves two steps: (1) recognizing a paraphrase (paraphrasing recognizer) and (2) reporting the final result (reporter). The *Paraphrase Recognizer* compares two internal representations (one is of a given sentence and another is of a student's input) and results in a paraphrase match (a "concept-relation-concept triplet" match), which also includes a paraphrase pattern. The *Reporter* provides the final result consisting of the total paraphrase matches, type of paraphrase matches, any missing information, and any extra information. Based on the similarity measure, this report will tell us whether the explanation is full or partial and whether it contains additional information.

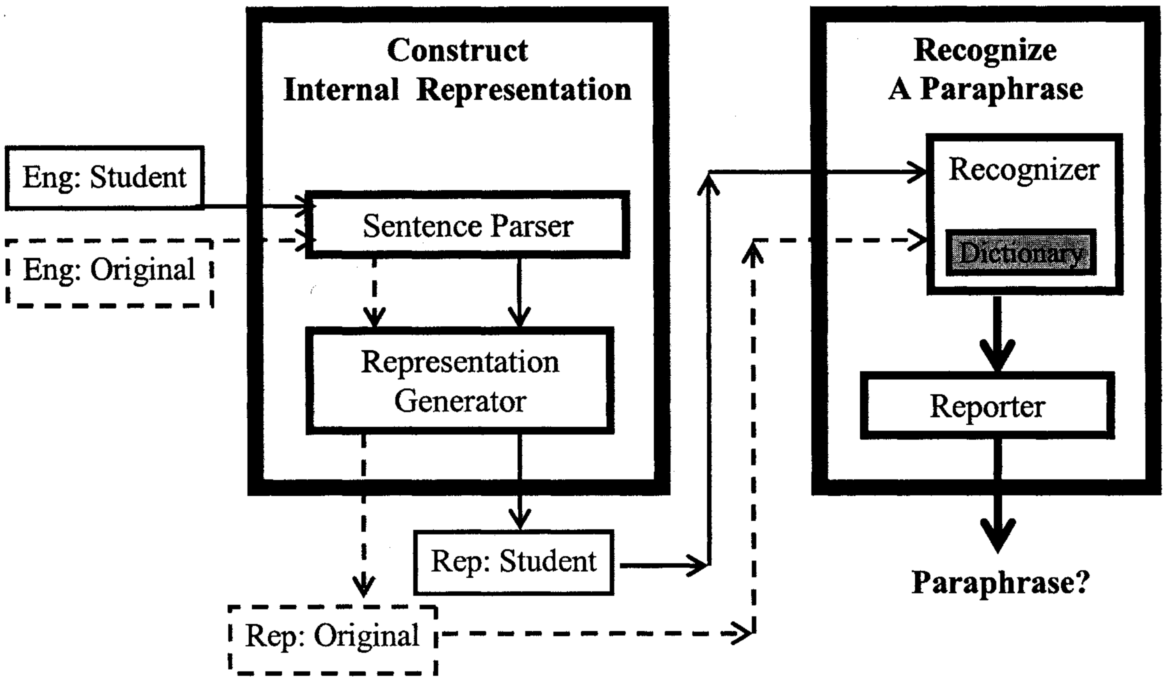


Figure 3: Architecture of the Recognition Process.

Outcomes of This Research

The main contributions of this work to the research community are the preposition sense classification and generalized disambiguation processes, which are integrated into the paraphrase recognition system and are shown to be highly effective. The recognition model is also a significant part of this contribution.

I achieved (1) modeling effective preposition classification based on their usage and designing the generalized preposition disambiguation module, (2) modeling the paraphrase recognition process and implementing a significant portion of this design demonstrating its effectiveness, (3) integrating the preposition disambiguation module into the paraphrase recognition system and gaining significant improvement, and (4) featuring the Syntactic-Semantic Graph as a sentence representation.

Outline Structures and Contents

The dissertation is organized as follows:

Chapter 2 contains background information and work related to this research. This includes sentence representations, paraphrase definitions, English sentence parsers, dictionaries and ontologies, word sense disambiguation (WSD), and preposition sense disambiguation (PSD).

Chapter 3 contains the paraphrase definition using during this research. This includes a number of challenges, such as sentence representation, paraphrase recognition, and paraphrase generation.

Chapter 4 describes the sentence representation "Syntactic-Semantic Graph" (SSG). Its features, a comparison with existing representations, and steps to constructing a SSG are also covered.

Chapter 5 contains the preposition classification based on usages. Each of seven general categories and specific usage-cases are described.

Chapter 6 describes the model to recognize paraphrases. For each paraphrase pattern, a model to recognize it is illustrated.

Chapter 7 contains the model for the preposition classification process. The integration of the preposition classification into the paraphrase recognition system is also described here.

Chapter 8 contains the experimental results.

Chapter 9 contains the analysis and discussion of results.

Lastly, Chapter 10 covers the conclusions and future work.

CHAPTER 2

BACKGROUND AND RELATED WORK

This section provides the background and the work related to this research. The first part deals with paraphrase definition, which is the starting point of this research. Then, a number of challenges related to building paraphrase systems are discussed, such as, how to represent a sentence, how to compare or evaluate two sentences. These challenges lead to the rest of discussions of the background work: sentence representation, English sentence parsers, dictionaries and ontologies, and sense disambiguation and classification.

Sentence Representations

The first challenge in building paraphrase systems is *sentence representation.* Selecting an appropriate representation is very important. A sentence has to be presented in a machine-readable format that the computer can read and process it. A simple representation (*e.g.* close to the natural language English sentence) may require more computer processing complexity and time. A more complex representation (*e.g.*, concepts and relation between concepts) will require more time in constructing a representation, but less time in processing it. Hence, a chosen representation will determine the complexity of each module in the system. One representation (*Syntactic Representation*) might describe a sentence in grammatical terms: subject, verb, object, modifiers etc. It might also include tense, mode, and voice. A semantic representation might contain conceptual relations among things or objects.

Logical Representation (Brna, 1999; Cawsey, 1994) uses the formulas of predicate logic to represent knowledge, an approach that is good for reasoning. Predicate logic is a development of *propositional* logic represented as an atomic proposition. Each proposition used in the system must be clearly defined: *predicate names* as well as a number of *arguments*, which may be *constant symbols* (*e.g.*, monkey, walnut), *variable symbols* (*e.g.*, X or Y), or *function expression* (*e.g.*, ancestor(monkey)). The logical representation is not suitable for the proposed system, since the students' inputs may be different than the system expected; consequently, there are no predicates to handle such input. Hence, the system becomes too restricted and limited.

Semantic Nets (Cawsey, 1994; Marshall, 2000; Wang, 1999) use graphs to represent concepts and relations. Each concept is described in terms of its relationship to other concepts, *e.g.* Mike is <u>*an instance of*</u> a person, and a person <u>*is a*</u> mammal. These relationships are described and represented in a *semantic network*, shown in Figure 4:

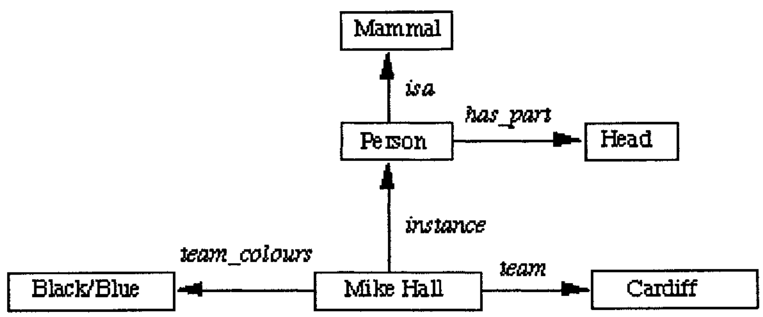


Figure 4:[4] A Semantic Network.

A semantic network for a sentence "John gave Mary the book" is as shown in Figure 5.

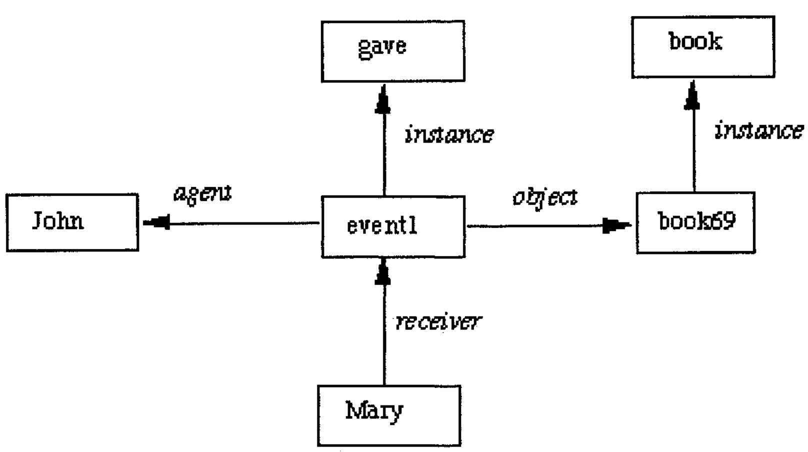


Figure 5:[5] A Semantic Network for a Sentence "John gave Mary the book."

[4] This image is retrieved from Marshall (2000).
[5] This image is retrieved from Marshall (2000).

Frames (Luger, 2002; Marshall 2000) represent a collection of attributes and associated values that describe an entity, e.g. a Person frame has an <u>*isa*</u> attribute containing a value <u>*Mammal*</u>. Each frame contains a number of slots and each slot is used for each attribute. Considered the example shown in Figure 4, Figure 6 illustrates how they are represented in Frames.

Semantic Network and Frame System are not suitable for the proposed system because relations and slots have to be defined prior to any use; hence, this makes the system limit to these defined rules (relations and slots).

Person	
isa:	Mammal
Cardinality:	...
Rugby-Player	
isa:	Person
Cardinality:	...
Height:	
Weight:	
Position:	
Team:	
Team-Colours:	
Mike-Hall	
instance:	Rugby-Player
Height:	6-0
Position:	Centre
Team:	Cardiff-RFC
Team-Colors:	Black/Blue
Rugby-Team	
isa:	Team
Cardinality:	...
Team-size:	15
Coach:	
Cardiff-RFC	
instance:	Rugby-Team
Team-size:	15
Coach:	T. Holmes
Players:	{R. Howley, M. Hall, ...}

Frames *Person, Rugby-Player* and *Rugby-Team* are classes.
Frames *Mike-Hall* and *Cardiff-RFC* are instances.

Figure 6: A Frame System.

Conceptual Dependency (CD, Schank, 1975; Marshall, 2000) uses four primitive conceptualizations to represent meaning of verbs: ACTs for action (*e.g.*, ATRANS for transfer of an abstract relationship, PTRANS for transfer of the physical location of an object), PPs (picture producers) for real word objects, AAs (action aiders) for attributes of actions, PAs (picture aiders) for attributes of objects, Ts for time, and LOC for locations. The CD representing the sentence "John gave Mary the book" is as shown in Figure 7, where arrows indicate the direction of dependency, double arrows indicate *two-way* links between the actor (PP) and action (ACT), and letters indicate certain relationships (*i.e.*, p=past tense, o=object, R=recipient-donor).

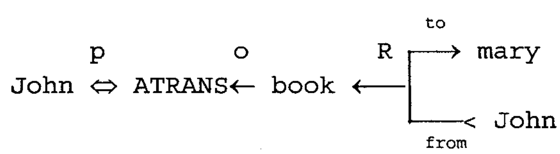
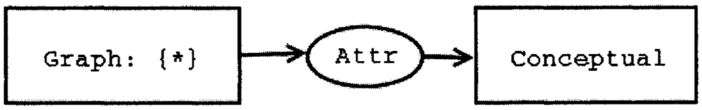


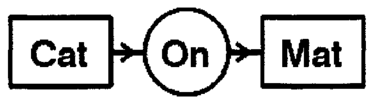
Figure 7: A Conceptual Dependency for a Sentence "John gave Mary the book."

Conceptual Graph (CG, Sowa, 1983; 1992) represents relations between concepts semantically. CG is a graph of two kinds of nodes: concepts and relations. The nodes have directed-arcs between them indicating relations, as shown in Figure 8 and Figure 9. A CG is a bipartite graph, that is, all arcs are only between concepts and relations. There are no arcs between two concepts and there are no arcs between two relations. In the linear notation, square-brackets '[ ]' are used around concepts and parentheses '( )'are used around relations.



Linear Notation: [Graph: {*}]->(Attr)->[Conceptual]

Figure 8:[6] A Conceptual Graph representing the phrase "Conceptual graphs."



Linear Notation: [Cat]->(On)->[Mat]

Figure 9[7]: A Conceptual Graph representing the sentence "A cat is on a mat."

Scripts (Schank & Abelson, 1977; Marshall, 2000) is a structured representation describing a stereotyped sequence of events in a particular context. A script includes the following components: *Entry Conditions* (must be satisfied before events in the script can occur), *Results* (conditions that will be true after events in the script occur), *Props* (slots representing objects involves in events), *Roles* (persons involved in the events), *Track* (variations on the scripts), and *Scenes* (the sequence of events that occur). An example in Figure 10 is a script describing a bank robbery.

Others: Some paraphrase generation systems (e.g. Stede's generation system (1996), Halogen) have designed their own representation appropriate for a sentence generation. Nevertheless, concepts of these representations are based on existing representations, such as frames and conceptual graphs.

[6] The image is retrieved from Sowa, http://www.jfsowa.com/cg/
[7] The image is retrieved from Sowa, http://www.jfsowa.com/cg/

Script: ROBBERY	*Track: Successful Snatch*
Props: G = Gun, L = Loot, B= Bag, C = Get away car.	*Roles*: R = Robber, M = Cashier, O = Bank Manager, P = Policeman.
Entry Conditions: R is poor. R is destitute.	*Results*: R has more money. O is angry. M is in a state of shock. P is shot.

Scene 1: Getting a gun

R PTRANS R into Gun Shop
R MBUILD R choice of G
R MTRANS choice.
R ATRANS buys G

(go to scene 2)

Scene 2 Holding up the bank

R PTRANS R into bank
R ATTEND eyes M, O and P
R MOVE R to M position
R GRASP G
R MOVE G to point to M
R MTRANS "Give me the money or ELSE" to M
P MTRANS "Hold it Hands Up" to R
R PROPEL shoots G
P INGEST bullet from G
M ATRANS L to M
M ATRANS L puts in bag, B
M PTRANS exit
O ATRANS raises the alarm

(go to scene 3)

Scene 3: The getaway

M PTRANS C

Figure 10:[8] A Bank Robbing Script.

[8] This image is retrieved from Marshall (2000).

Paraphrase

What is a Paraphrase?

What is a paraphrase? The answer starts off with "*paraphrase is a restatement or a way to talk about the same situation in a different way*", although "the same situation" and "a different way" can be interpreted in different ways (Hurst, 2003).

Academic writing centers (ASU Writing Center, 2000; Quality Writing Center, 2002; BAC Writing Center, 2002; USCA Writing Room, 2002) have a common characterization of "*paraphrasing means restating ideas in our own words*". This can be achieved by exchanging the original words with our own words. We can use synonyms or different word forms or change the sentence structure to create our own rhythm. An example (from The Quality Writing Center, University of Arkansas, 2002) of paraphrases of the opening sentence of the Gettysburg Address by Abraham Lincoln:

> *Original*: Four score and seven years ago, our fathers brought forth on this continent a new nation, conceived in liberty and dedicated to the proposition that all men are created equal.
>
> *Use of Synonyms*: Eighty-seven years before now, our ancestors founded in North America a new country, thought of in freedom and based on the principal that all people are born with the same rights.
>
> *Restructuring the Sentence*: Our ancestors thought of freedom when they founded a new country in North America eighty-seven years ago. They based their thinking on the principle that all people are born with the same rights.

In addition to those defined in previous paraphrase characteristics, Hawes (2003) stated that a brief definition or an example is also a part of paraphrasing. Hence, a paraphrase sentence may be longer than the original one. According to McNamara (2004), using definition or examples which include knowledge outside the text is considered to be an *elaboration* rather than a paraphrasing. The proposed system will use a brief definition as a part of paraphrase, but not an example.

Stede (1996) says "*if two utterances are paraphrases of one another, they have the same content and differ only in aspects that are somehow secondary*". A question can be asked on how to interpret "the same content" and "secondary aspects."

In summary, different authorities uses different paraphrase definitions and some definition may raise more questions, *i.e.*, how to interpret "same situation", "the same content", "different ways.:" The academic writing centers provides a distinct set of paraphrase patterns and that are the most useful. Hence, the definition of a paraphrase in this research is based on the academic writing centers covering the usage of definition (Hawes, 2003). The detailed of definition is described in Chapter 3.

Paraphrase Challenges

There are number of challenges involved in building paraphrase recognition systems. The first and the biggest issue is the representation of a sentence. How to represent a sentence and the meaning of the sentence? Will the syntactic structure be sufficient? Or is a semantic structure required? The detailed discussion of existing *sentence representations* is described in the following section. Once a sentence and its knowledge are represented, the next issue is *recognizing paraphrases.* How to recognize the similarity between two sets of sentences (a set may contain one or more sentences) or two representations? Are these two representation paraphrases of one another? The recognition model has to *measure paraphrase distance* – how different or similar these two sentences are? – and to *explicate the differences between various paraphrase patterns*. The distance can be measure using the concept-relation matching pairs. If the pair is match between two representations and if that relation is in the high-weight (*e.g.*, Agent, Patient, details in next section) set, then the distance will be impact more by this match. If the relation is in the low-weight set (*e.g.*, Article, Modifier), then distance will receive fewer impact. The system has to differentiate these two matches: high-weight vs. low-weight. If the system involves constructing a sentence (*i.e.*, the machine translation) then *generating a paraphrase* is one of the major concerns. Depending on the size of and kind of the applications, different challenges have to be overcome to achieve the application goals.

How do other researchers recognize paraphrases?

A number of people have worked on paraphrase recognition. This section briefly describes some of those works primarily to illustrate different ways to implement a paraphrase recognition system and that they are application-specific and why it will not work for the proposed system.

AutoTutor (Graesser et al., 2000; 2001) is a computer-based tutor developed by the Tutoring Research Group at the University of Memphis. This system simulates a typical human tutor having a conversional dialog with the student. For each question in a lesson, ideal answers and anticipated bad answers are included in the curriculum script. Once a student answers a question, the answer is passed through language analyzers that use Latent Semantic Analysis (LSA; Landauer, Foltz, & Laham, 1998) to assess the coverage of the ideal answers. AutoTutor uses LSA to analyze the student's input with relative success. One problem is that LSA uses the concept 'bag of words', which means that any word found in the LSA matrix space will contribute to the final result. Similarly, a lack of words in the "bag of words" also impacts the final result. This LSA deficiency was also found in the iSTART evaluation system when the only LSA was used[9].

CIRCSIM-Tutor, a tutoring system, (Glass, 2001; Cho et al., 2000) focuses on the student understanding the topic using a short-answer dialogue. The expected answers are very short and even if the student's answer is long, the system will look for just that short expected answer to see whether or not it was covered using simple word matching. Their main goal is to understand human tutoring and to discover which tutoring strategy gives the best result. CIRCSIM cannot handle all student answers due to its lack of understanding of meaning.

DIRT (Discovering Inference Rules from Text; developed by Lin and Pantel, 2001a; 2001b) is an algorithm that uses inference rules in question answering and information retrieval. They use Minipar as a sentence parser, whose output is a *dependency tree*. For example, for "John found a solution to the problem" a path between a node "John" and node "problem" is "N:subj:V ← find →V:obj:N → solution → N:to:N", which generally means "X finds solution to Y". To find a paraphrase of this

[9] Details on the comparison among 8 different iSTART feedback systems can be founded in McNamara, Boonthum, et al. (2006).

means finding a different path of this sentence (from its dependency tree) between the same words, *i.e.,* a different path starting from a node "John" to a node "problem."

The *ExtrAns* (Extracting Answers from technical texts) question-answering system by Molla et al. (2003) and Rinaldi et al. (2003) uses minimal logical forms (MLF; that is, the form of first order predicates) to represent both texts and questions. They identify *terminological* paraphrases by using a term-based hierarchy that includes synonyms and variations; and *syntactic* paraphrases by constructing a common representation for different types of syntactic variation via meaning postulates. In the absence of a paraphrase, they loosen the criteria for identifying this paraphrase by using hyponyms, finding the highest overlap of predicates, and simple keyword matching.

Barzilay and Lee (2003) also identify paraphrases in their paraphrased sentence generation system. They first determine different paraphrasing rules by clustering sentences in comparable corpora using *n*-gram word-overlap. Then for each cluster, they use *multi-sequence alignment* to find intra-cluster paraphrasing rules: either morpho-syntactic or lexical patterns. To identify inter-cluster paraphrasing, they compare the slot values without considering word ordering.

C-Rater under development at ETS by Leacock and Chodorow (2003) is a system that scores short-answer questions by analyzing the conceptual information of an answer in respect to the given question. Since C-Rater is designed to measure a student's understanding of specific content material (Leacock, 2004), effort from content experts (test developers or teachers) is required to develop "*gold standard*" responses. The student's answer is compared to the correct answer in effect recognizing a paraphrase between the two. The predicate argument structure is used to represent the two answers and the matching is rule-based. The developers report that the scoring system seems to work, but that a confidence of score cannot be indicated, so this scoring cannot be used to grade the answer! Instead of the rule-based approach, a statistical version of C-Rater is being developed by Thomas Morton using probability to indicate the system's confidence (Leacock, 2004).

Uzuner et al. (2005) proposed using low-level syntactic structure to identify plagiarism. They detect creativity of writing and linguistic similarities, such as structures of sentence-initial and -final phrases and verb classes. Their recognition process contains the following features: *TFIDF-weighted keyword* (term-frequency inverse document-frequency; frequently use of keywords), *feature words* (special words or terms describe), *distributions of word lengths and sentence length*, and *baseline linguistic features* (a set of surface, syntactic, and semantic features).

Qiu et al. (2006) proposed a paraphrase recognition system based on dissimilarity, rather than similarity measurement, although both measurements are used. Their system is called *two-phase paraphrase recognition*: first phase is *Similarity Detection* and second one is *Dissimilarity Classifier*. They use *predicate argument tuples* (a structure representing a verb and its arguments) to contain information on a sentence's action, concepts, and the relationship among them. In Phase I, they compare a pair of tuples to detect a similarity. The tuples remaining unpaired from Phase I will be labeled by Phase II of their signification. That is, some extra information may be important, while others may not be useful information.

As described in this section, there are different ways to recognize paraphrase, from a simple word matching to a deeper semantic comparison. My research is similar to C-RATER in the way that both systems convert a natural language sentence into a semantic representation and find coverage between two sentences. One of differences between these two is that C-RATER requires experts to identify the ideal answers before that question can be used. The systems developed after year 2003 are presented in this section so as to endorse the significance of this work. That is, recognizing a paraphrase is important and worthwhile, and there is a large amount of on-going research on this.

English Sentence Parsers

Prior to converting a natural language sentence into a representation (one of representations described in previous section), a sentence has to be parsed and its syntactic structures (*i.e.*, par-of-speech of each word, subject, object, verb) identified. Hence, a *sentence parser* is needed. There are a number of English sentence parsers available, but only three of them have been investigated in detail and are briefly described below:

Link Grammar, developed at Carnegie Mellon University (2000), is a syntactic parser that assigns to a sentence a syntactic structure that consists of a set of labeled links connecting pairs of words. Valid word use is represented by rules about how each word may be linked to others. A valid sentence is one in which all words are connected with valid connecting rules. Thus a parse is a solution to the problem of finding links to connect all the words in the sentence. The parser is able to skip some portions of the sentence that it cannot understand and able to handle some unknown vocabulary. An advantage of using this parser is that it gives all possible solutions for a sentence.

Minipar is a broad-coverage parser developed by Dekang Lin (2003; 2001a; 2001b) during his work on DIRT - Discovery of Inference Rules from Text. Minipar represents the grammar as a network, where the nodes represent grammatical categories and the links represent types of syntactic (dependency) relationships. The grammar is manually constructed and the lexicon is derived from WordNet (see below), plus some additional proper names. Each word has all of its possible part-of-speech uses in its lexical entry. To construct a parse for a given sentence, Minipar finds all possible parses using its grammar; however, it will show only the highest-ranking output. (The ranking is based on the statistics obtained by parsing a sample corpus with Minipar.) Lin claims that Minipar is very efficient and his evaluation on the SUSANNE corpus shows that MINIPAR achieves about 88% precision and 80% recall with respect to dependency relationships. Minipar has the benefit of grouping words together, e.g. "life history". Naturally, these group-compound words have to be defined in the Minipar dictionary. Therefore, to cover more grouping words, we are allowed to add words in the Minipar dictionary.

Connexor (2002) is a commercial parser product that tags each word with its word position, base-form (or lemma), functional dependency, functional tag, surface-syntactic tag, and morphological tag. Like Minipar, Connexor outputs only one parse result. Although most of the parse results are reasonable, there are a number of common cases where Connexor gives an incorrect parse. For example, it cannot properly handle a complex sentence, containing a coordinator (*e.g.*, and, or).

For this work, the Link Grammar parser has been chosen for a number of reasons: (1) its ability to produce several possible parse results, (2) the results are ranked according to likelihood, and (3) the output from Link Grammar is in the form of triplets, which is similar to the chosen semantic representation and so simplifies the mapping between syntactic and semantic structures.

Dictionaries and Ontology

This section discusses different dictionaries and ontologies. Most of these are inventory resources for all part-of-speech words (*e.g.*, dictionary) whereas some are mainly for nouns or verbs (*e.g.*, WordNet). Dictionaries and Ontologies[10] play a big role in paraphrase recognition, especially in determining relations among words (such as synonym).

WordNet, developed by the Cognitive Science Laboratory at Princeton, is one of the electronic lexical resources most used in NLP applications (Miller et al., 1993; Fellbaum, 1998). It contains English nouns, verbs, adverbs, and adjectives but its focus is on nouns more than other kinds of part-of-speech (*i.e.*, verbs, adjectives, and adverbs). Words are grouped together in WordNet if they are related to one another in one of the following ways: synonym, hypernym (is-a, a more generic term), hyponym (a more specific term), antonym, troponym (a manner of doing something), coordinate term, sentence frame, or familiarity. The latest version 2.1 contains over 155,000 words and 207,000 word-sense pairs (*i.e.*, synsets. A word will have a number of synsets, each synset means one sense of such word and it contains a list of words that can be used interchangeably for that word's sense).

FrameNet (UC Berkeley, 2000) is an on-line lexical resource for English, based on frame semantics and supported by corpus evidence. A word is organized in a frame format rather than by its lemma. For example, "bake" is defined under an "Apply_heat" frame, which describes a situation that involves a Cook (a person does the cooking), some Food, and a Heating_Instrument (*e.g.*, oven). FrameNet also organizes words in a hierarchy (Is-A relation). The current FrameNet lexical database contains more than 8,000 lexical units (pairs of a word with a meaning), more than 6,100 of which are fully annotated, in more than 625 semantic frames, exemplified in more than 135,000 annotated sentences. Although the FrameNet database uses well-defined annotation, it is not readily usable as an ontology. More words, especially nouns, have to be denoted.

[10] In philosophy, the word "ontology" refers to the subject of existence. In AI, an "ontology" is a specification of a representational vocabulary for a shared domain of discourse -- definitions of classes, relations, functions, and other objects (Gruber, 1993).

Cyc (Cycorp, 2002) is the Very Large Knowledge Base (VLKB) developed by Doug Lenat at MCC (Microelectronics and Computer Technology Corporation, now Cycorp, Inc.). Cyc captures the common sense knowledge (both implicit and explicit knowledge) in a hundred randomly selected articles in the Encyclopedia Britannica and contains over 1.5 million *"facts, rules-of-thumb and heuristics* for reasoning about the objects and events of everyday life" (Cycorp, 2002). It uses a first-order-predicate calculus with extensions for terms and assertions needed to be used in describing the Cyc Knowledge-Base. The extensions are used to handle equality, default reasoning, skolemization and some second-order features. Though Cyc appears to be a good knowledge inventory, the ways that predicates are defined in Cyc make it difficult to use.

The *Longman Dictionary of Contemporary English* (LDOCE; Longman, 2005) is one of the most widely used dictionaries in language research. The latest 4th edition contains 155,000 natural examples, 88,000 new spoken example sentences, 1 million additional sentences from books and newspapers, and 4,000 new words and meanings. LDOCE has an online version; however, it is still represented in a traditional way. That is, definitions are in natural language sentences or phrases and only synonyms are listed.

Roget's Thesaurus of English Words and Phrases is a collection of words and phrases. According to Roget (1852), "… a collection of the words the English language contains and of the idiomatic combinations peculiar to it, arranged, not in alphabetical order as they are in a Dictionary, but according to the ideas which they express …" The Penguin edition by Betty Kirkpatrick (1998) consists of six classes, 990 headwords, and more than 250,000 words. The word classification in this edition is similar to that of the original edition in 1852. This dictionary would be useful for recognizing paraphrase using idiomatic expressions.

WordNet is chosen for this work because of it well-structured electronic lexical resource that provides not only word meanings (a feature of dictionaries), but also relations among words beyond synonym lists (features of ontologies).

Word Sense Disambiguation in General

What is a word sense?

A word, given its part-of-speech, usually has a default or primary or intuitive meaning. For example, when the noun "house" is mentioned, it is usually in reference to "a residence or place in which people live" while the verb "house" means "to provide someone with a place to live." However, when a word is put in a particular context, the meaning may be changed from the default, depending on the surrounding words. For example:

(1) a. John builds a *house.*
  b. John is a member of *the House of Representatives.*
  c. John performs in a *vaudeville house.*
  d. John *houses* twenty foreign visitors.
  e. The Science Museum *houses* the Asia Art Collection.
  f. John buys *house paint.*
  g. John orders *the house wine* at the restaurant.
  h. John's performance *brings down the house.*
  i. Drinks are *on the house.*

The word "house"[11] in (1.a) is a noun that means "a residence"; in (1.b), "a body of a legislature"; and in (1.c), "an auditorium." The verb "house" can mean "to provide with a place to live" in (1.d) or "to keep something in that place" in (1.e). The (1.f) and (1.g) are examples of an adjective "house" that means "suitable for a house" and "served by a restaurant as its customary brand", respectively. The last two examples are "house" in idiomatic expressions: "highly successful" in (1.h) and "free" in (1.i). As can be seen, the word "house" alone can be used in at least four different ways (noun, verb, adjective, and idiom) and has at least seven different meanings. Each variation of meanings is

---

[11] The meanings of "house" are retrieved from *Longman Dictionary of Contemporary English (Online)* http://www.ldoceonline.com/

based on a part-of-speech and surrounding words. That is, a word is *context-sensitive*. Each of these meanings is defined as a "*word sense.*"

So, *word sense disambiguation* (WSD) is a process to find a *meaning* of a *word* in a given *context* (Agirre & Edmonds, 2006). The computational difficulty for WSD is how to describe logically the thought-process or the human-way of disambiguating, which can then be computerized. WSD research was first used in machine translation in the late 1940s, matching a word from one system (or language) to a word in another system (or language), but by the late 1970s, WSD had become an artificial intelligence (AI) research topic, that of natural language understanding. The next section briefly describes different WSD approaches.

Basic Approaches to WSD

Approaches to WSD are often classified according to the source of information used in differentiating one sense from another (Agirre & Edmonds 2006). *Knowledge-based* (or *dictionary-based*) approaches are methods that mainly use dictionaries, thesauri, and lexical knowledge bases whereas *corpus-based* approaches are methods that use a corpus (a collection of texts or sentences) to learn and train the system on sense discrimination

Knowledge-based approaches have been studied by many researchers including Lesk (1986), Cowie et al. (1992), Wilks et al. (1993), and Rigau et al. (1997), who all used machine-readable dictionaries (MRDs); Agirre and Rigau (1996), Mihalcea and Moldovan (1999), and Magnini et al. (2002) used WordNet. Lesk (1986) derives the correct word sense by counting word overlap between dictionary definitions of the words and the context of the ambiguous word while Wilks et al. (1993) use co-occurrence data extracted from an MRD to construct word-context vectors (*word-sense vectors*). The Noun-WSD by Agirre and Rigau (1996) was created using the WordNet noun taxonomy and the notion of *conceptual density* by measuring a conceptual distance between two concepts as the length of the shortest path that connects the concepts in a hierarchical semantic net. The final result yields the highest density for the sub-hierarchy containing more senses of those, relative to the total amount of senses in the sub-hierarchy. Mihalcea and Moldovan (1999) disambiguate nouns, verbs, adjectives and adverbs using WordNet senses while Magnini et al. (2002) focus on the role of domain using WordNet domains. In summary, these WSD methods are in the same class because they are *knowledge-based* and use *structured lexical knowledge resources*. Yet they differ in the *lexical resource used* (*e.g.*, MRD or WordNet), *the* information *contained in this resource* (*e.g.*, senses, taxonomy, co-occurrence), and *the property used to relate words and senses* (*e.g.*, overlap words appeared in a dictionary definition versus those appeared the context of the ambiguous word).

Corpus-based approaches utilize statistical and machine-learning (ML) techniques to train the system in WSD. A number of ML techniques have been studied including decomposable model (Bruce & Wiebe, 1994; using a subclass of log-linear models to characterize and study the structure of data in the corpus, *i.e.* interactions among words and their co-occurrences), Maximum Entropy (Suarez & Palomar, 2002; estimating probability distributions and selecting the distribution that maximizes entropy and satisfies the constraints imposed by training data), decision lists (Yarowsky, 1994; identifying patterns, collecting data from the corpus, measuring collocation distributions and sorting them by log-likelihood), neural networks (Towell & Voorhees, 1998; using nodes to represent words and concepts and links to represent their semantic relations), support vector machines (Cabezas et al., 2001, Lee et al., 2004; finding the hyperplane that uses the information encoded in the dot-products of the transformed feature vectors as a similarity measure.), and distribution estimation (Chan & Ng, 2005; estimating sense distribution and priori probabilities of senses).

Combinations of existing methods are being investigated, such as combining a specification marks methods (SM, one of the knowledge-based methods) and a maximum entropy-based method (ME, one of the corpus-based methods) to disambiguate noun sense (Montoyo et. al, 2005). It is worth noting that all of these efforts are for disambiguating nouns, verbs, adjectives, and adverbs. They do not, to my best knowledge, cover the disambiguation of prepositions.

Existing Corpora

In addition to the lexical word-sense resources (*e.g.*, dictionaries), the corpus-based WSD systems require a collection of texts or sentences to complete their tasks. Hence, *Corpora* with and without annotation are constructed. This section describes the three corpora most used by the above WSD systems.

The Brown Corpus (Francis & Kucera, 1964) is a million-word "balanced" collection of texts containing samples of writing prose and classified into 15 categories: reportage, editorial, reviews, religion, skill and hobbies, popular lore, belles-lettres, learned, fiction general, mystery and detective fiction, science fiction, adventure and western fiction, romance and love story fiction, humor, and miscellaneous. All together, there are about 500 texts; each contains about 2,000 words. Experts annotated a subset of these texts and sentences using part-of-speech tags defined in the Penn Treebank.

The British National Corpus (BNC: BNC, Consortium, 2001; Burnard, 2000; Leech, 2000) is a reasonably balanced corpus. It contains more than 4,000 samples of contemporary British English and contains more than 100 million words. The corpus is encoded using ISO standard 8879 (SGML: Standard Generalized Markup Language) to represent both the output from the automatic part-of-speech tagger (called *CLAWS* developed by Roger Garside at Lancaster) and a variety of other structural properties of texts (*e.g.*, headings, paragraphs, lists).

The Wall Street Journal Corpus (WST; Paul & Baker, 1992) contains almost 40 million words from *Wall Street Journal* articles from 1987 through 1990. It is the base of the manually annotated DSO (Defense Science Organization of Singapore), Penn Treebank, and PropBank corpora.

Sentences from the Brown Corpus are used in the present work since they are available at no charge.

Preposition Sense Disambiguation

As mentioned above that most of the word sense disambiguation has been put for disambiguating nouns, verbs, adverbs, and adjectives and none of them raises issues about prepositions. One of the main contributions of this research is on preposition disambiguation: preposition classification and generalization disambiguation process.

This section will also demonstrate that the preposition disambiguation is currently an active research and how to benefit it in the proposed paraphrase recognition system.

Differences of PSD from WSD

The meaning of a preposition is different from the meaning of a noun, verb, adverb, or adjective. That is, disambiguating a noun involves finding a correct synonym or definition, but finding the correct sense of the preposition is not finding a synonym that could be substituted; it is identifying the relation between the two things that the preposition connects. To find this relation, there are two main steps: (1) identify which word the preposition is attached to (called *prepositional phrase attachment*), and (2) determine how two things or concepts which the preposition connects (called *a relation*) should be interpreted. For example, a meaning of "with" can be identified by first identifying its attachment – either to a noun or a verb. Let's say, "with" is attached to a noun, then the default meaning of the preposition "with" is that "two things are together." That is, it indicates this *relation* between the two things the preposition connects. Similarly, the preposition "to" has a default usage indicating "a destination." However, when these prepositions appear in a context, the default may no longer apply.

(2) a. John builds a house *with* Tom.
 b. John builds a house *with* a hammer.
 c. John builds a house *with* passion.
 d. John builds a house *with* a kitchen.

(3) a. Mary goes *to* school.
 b. Mary works from 9 to 5.
 c. Mary loves *to* dance.
 d. Mary sits next *to* John.

The preposition "with" in (2.a) has the default usage to indicate "two together", that is, "John and Tom together"; while in (2.b) "with" indicates "an instrument"; in (2.c), "a manner"; and in (2.d) ,"a part of or attribute." Similarly in (3), the preposition "to" can indicate "a location destination" (3.a); "a time destination" (3.b); "an action/ intention" (3.c); or "a point location/direction" (3.d). A preposition is also *context-sensitive*. These so-called preposition's meanings are in-fact preposition's usages; hence, a preposition sense disambiguation (PWSD) is a process to identify a *usage* of a *preposition* in a given *context*.

Approaches to Preposition Sense Disambiguation

On the one hand, approaches to PWSD are similar to those for WSD regarding resources: *knowledge-based* vs. *corpus-based*. However, a selected approach depends on what kind of disambiguation or classification of a preposition needs to be solved. If the problem is one of structural ambiguity, that is deciding which part of the sentence the prepositional phrase is augmenting, then the disambiguation process is Prepositional Phrase Attachment (PPA). *Knowledge-based* approaches to PPA include *syntactic or lexical cues* (Wu & Furugori, 1996), *syntactic and semantic features* (Mohanty et al. 2005). *Corpus-based* approaches PPA include *co-occurrence* (Wu & Furugori, 1996), probabilistic models such as *Maximum Likelihood Estimation* (MLE, Kayaalp et al., 1997). Some work has been done using a combination of knowledge-based and corpus-based include Merlo and Leybold (2001) and Mitchell (2004, through instance-based learning).

On the other hand, disambiguating or classifying a preposition can be different from WSD. Certainly once we solve the PPA, the next question regards what that preposition indicates. That is, what is the purpose for which that preposition is used? One might say that this is a chicken-and-egg problem, that is, to solve PPA, we need to know the purpose of the preposition. Then, to find the purpose for which the preposition is used, PPA has to be resolved. Then this will be an cyclic problem.

Bannard and Baldwin (2003) attempted to capture the semantics of prepositions in terms of a transitive property (either a preposition is intransitive or transitive). This seems to be similar to the PP-attachment problem, but they capture the *verb-particle* (verb and preposition together constitute one meaning).

Alam (2004) has worked on the disambiguation of 'over' by considering its meaning with respect to two main categories: one is the meaning that can be identified by its complement noun phrase and the other is the one that derives from its head (verb or noun phrase). Alam defines the subcategories of 'over' in terms of the various features of head and complement. For both head and complement, ontological categories (hypernyms) are used, *e.g.*, furniture is a *physical object*, coffee is a *drink*. Two decision trees are proposed: one for the head and another for the complement. To determine the meaning of 'over', the complement decision tree is examined first. Alam claimed that this is because the meanings of 'over' can be identified mostly from its complement. If the sense of 'over' cannot be identified by this tree, then the head decision tree is checked. Alam performed this evaluation manually, though she claimed that the algorithm should be easy to implement.

Harabagiu (1996) used WordNet to disambiguate prepositional phrase attachments. She used the hypernym/hyponym relation of either verbs or nouns or both from WordNet to categorize the arguments of preposition relations. This approach is based on inferential heuristics. Three heuristic rules were defined for the preposition 'of' in order to understand different types of valid prepositional structures.

Mohanty et al. (2004, 2005) have used preposition syntactic frames to define prepositional semantics. A number of rules are defined to analyze each frame type (attribute of a verb, attribute of a noun before a preposition and a noun after a preposition) to disambiguate prepositional phrase attachment as well as to identify the semantic relation of this attachment. They used a system called UNL (Universal Networking Language), which has its own lexical knowledge. Its English Analyzer uses both the existing English grammar rules in UNL itself and user-defined rules of preposition attachment and semantics, and then generates a UNL expression. They began with the preposition 'of' and expanded this concept for other prepositions (*for, from, in, on, to, with*).

It is worth noting that this dissertation has applied, among other things and with some improvement, the ideas of using features of the head and complement based on Alam's work (2004) and using WordNet ontological categories (although for different purposes) following Harabagiu (1996).

Preposition Classification Inventories

Like other words, the meaning of prepositions should be defined and can be found in dictionaries. In addition, meanings of prepositions or rather usage of prepositions are defined in grammar books. Therefore, this section lists a number of resources covering usage of prepositions.

Dictionaries: In addition to *LDOCE* (described above), Oxford English Dictionary is the most definitive reference book. *Merriam-Webster Online* (Merriam-Webster Inc. 1997) is based on the print version of *Merriam-Webster's Collegiate Dictionary, Eleventh Edition*. It contains over 165,000 word-entries and 225,000 definitions, including 10,000 new words and phrases and 40,000 usage examples. Another online dictionary is *Dictionary.com*, developed by Lexico Publishing Group LLC (1995). It is a multi-source dictionary that allows the user to look up the word meanings. The dictionaries appeared on this site include: Random House Unabridged Dictionary, The American Heritage Dictionary of the English Language and of Idioms, Webster's Revised Unabridged Dictionary, WordNet 2.0, Online Medical Dictionary, Merriam-Webster's Dictionary of Law, and Merriam-Webster's Medical Dictionary.

A comprehensive Grammar of English (Quirk et al., 1985) contains the information that describes how each preposition is used to denote one or more of the following relations: spatial relations: *dimension* (line, surface, area) such as destination, source, space, passage, movement, orientation; *time* such as time position, duration, before/after, since ... until cases, between ... and cases, by; *cause/purpose spectrum* such as cause, reason, motive, purpose, recipient, goal, target, origin; *means/agentive spectrum* such as manner, instrument, agentive, stimulus, accompaniment, support, opposition; and *other meanings*, which include miscellaneous cases that do not fall in the previous four, such as having, concession, various relations indicated by of, etc.

Lexical Structure. Jackandoff (1983, 1990) defined six positions of how prepositions can be used (*Spatial/Location, Temporal, Possession, Identification, Circumstance,* and *Existence*) while Dorr's LCS (2001) includes ten positions (four positions have been added to Jackandoff's: *Intention, Perception, Communication,* and *Instrument*). Only six positions are relatively close to the conceptual usages of prepositions: *Location, Possession, Intention, Instrument, Identification,* and *Temporal.*

The Preposition Project (TPP, Litkowski & Hargraves, 2005, 2006) is an ongoing project to construct a preposition ontology based on the definitions in Quirk et al. (1985). For each preposition, each sense contains a well-defined FrameNet (University of California, Berkeley) instance; however, the properties of complement and attachment in TPP are English phrases (e.g. "permeable or breakable physical object", "a perceived object; sometimes complement of a verb of perception") which would require an additional parser or transformation before it would be usable. The following prepositions have been completed while the rest are under development: about, against, at, by, for, from, in, of, on, over, through, to, and with.

Preposition Case-Marker (Barker, 1996) is a set of roles that prepositions can be used to indicate a relation. This set contains the following markers: *Accompaniment, Agent, Beneficiary, Exclusion, Experiencer, Instrument, Object, Recipient, Cause, Effect, Opposition, Purpose, Direction, LocationAt, LocationFrom, LocationThrough, LocationTo, Orientation, Frequency, TimeAt, TimeFrom, TimeThrough, TimeTo, Content, Manner, Material, Measure,* and *Order*. Each preposition covers a number of case-markers, indicating in which cases or situations the preposition can be used.

Table 1 shows the number of preposition classifications from these different resources for the prepositions I investigated, the ten most frequently used prepositions in the Brown corpus (Edict VLC, 2004): *of, to, in, for, with, on, at, by, from,* and *over*. These prepositions cover 85.63% of all the occurrences of the 46 prepositions used in this corpus. The Brown corpus consists of 1,015,945 words, of which 14.2% are prepositions.

Cyc (Cycorp, 2002), as mentioned in the previous section, could not easily be used as a preposition inventory. Unlike others, Cyc does not define a preposition based on its usage, but rather based on its truth predicates describing sentences and/or situations. Cyc predicates are not specifically used as relations between 2 components. Its predicates can have more or less than 2 arguments. In most cases, prepositions together with verbs (*verb-particles*) are defined as predicates. Cyc defines the true meaning prepositions after resolving the attachment (*i.e.*, verb-particles) or a set of arguments. To utilize Cyc KB, the preposition classification must be applied first.

Resources	of	to	in	for	with	on	at	by	from	over
Longman's Dictionary	18	20	15	25	15	21	16	20	16	13
Merriam-Webster online	12	8	5	10	11	10	6	11	3	7
Dictionary.com	21	16	9	10	27	13	13	11	5	14
Lexical Conceptual Structure (LCS)	3	10	24	7	14	7	6	11	17	6
The Preposition Project (TPP)	18	17	11	14	16	23	12	22	14	16
Quirk	4	5	3	7	8	6	6	6	8	11
Barker	3	9	12	11	8	7	7	10	4	6

Table 1: A count of preposition meanings from different sources.

CHAPTER 3

PARAPHRASE DEFINITION

This chapter describes the paraphrase definitions and the challenges encountered during this research.

Paraphrase Definition

Instead of attempting to find a single paraphrase definition, I have begun with six commonly mentioned paraphrase patterns. An Application based on this approach could allow the activation or deactivation of these patterns according to the user's needs; hence, these patterns accommodate various definitions.

Synonym Substituting a word with its synonym is one of the easiest ways to paraphrase, for example, the verb "help" can be replaced by its synonyms "assist" or "aid" (as shown in Figure 11)

John *helps* Mary. John *assists* Mary. John *aids* Mary.

Figure 11: Examples of using synonyms.

This pattern also covers other kinds of word relationships, such as antonym (opposite meaning), hypernym (a more generic term), hyponym (a more specific term), meronym (a part of a larger whole), and holonym (a whole of which a given word is a part).

John *hates* Mexican food.

John *does not like* Mexican food.

Figure 12: Examples of using an antonym with negation.

John catches a *bird*.

John catches a *cock*. (male bird)

Figure 13: Examples of using hypernym / hyponym.

Voice Changing the voice of sentence from active to passive or vice versa is considered paraphrasing although the focus of a sentence is changed from the *Agent* (doer of an action) to the *Patient* (receive an action).

<u>John</u> *helps* Mary.

<u>Mary</u> *is helped by* John.

Figure 14: Examples of changing voices.

Word-Form or Part-of-speech Changing a word into a different form, such as changing a noun to a verb, adverb, or adjective creates another paraphrase pattern. Depending on how the part-of-speech has been changed, the structure of the sentence may be changed.

John *makes changes to* the program. (changes = noun)

John *changes* the program. (changes = verb)

Figure 15: Examples of changing part-of-speech that does not affect the sentence structure.

John *uses* a hammer to build a house. (uses = verb)

John builds a house *using* a hammer. (using = gerund)

Figure 16: Examples of changing part-of-speech that does affect the sentence structure.

Breaking A Sentence or Combining Sentences A long and complex sentence can be broken into smaller simple sentences and still maintain the same description of the situation. Similarly, a number of small sentences can be combined to create a long sentence; yet, preserve the same information of the situation.

Mary is a high-school teacher. She teaches English.

Mary is a high-school teacher and teaches English.

Mary is a high-school English teacher.

Figure 17: Examples of breaking a sentence or combining sentences.

Definition/Meaning A word can be substituted with its definition or meaning. This is not only to create a paraphrase sentence, but also to explain and simplify a situation description. An example in Figure 18 uses a definition of "history", which means "the continuum of events occurring in succession leading from the past to the present and even into the future" (from WordNet 2.0). Even though the paraphrase sentence does not use exact definition, it does cover the key points of "history". That is "beginning", "go through", and "end."

All thunderstorms have a similar life *history*.

Thunderstorms go through similar cycles. They will begin the same, go through the same things, and end the same way.

Figure 18: Examples of using a definition.

Sentence Structure. The same situation can be stated in a number of different ways. There are many ways of saying "There is someone happy", for examples, "Someone is happy", "A person is happy", and "There is a person who is happy." Basically, different sentence structures are used to express the same thing. This normally involves in other paraphrase patterns, such as changing part-of-speech and changing voice, as shown in Figure 19.

John *uses* a hammer to build a house.

There is a hammer that John *uses* in building a house.

John builds a house *using* a hammer.

A house is built by John *using* a hammer. (voice)

Figure 19: Examples of using different sentence structures.

Challenges

There are four main issues covered in this work. First, *a sentence representation*: how should each sentence and knowledge about the sentence be represented? Instead of choosing existing representations (either syntactic or semantic), I have chosen to use a representation that combines syntactic and semantic representations. The semantic representation describes the situation and in a paraphrase recognition system, paraphrases describe the same situation; hence, they have the same semantic representation. However, having only semantic representation loses a number of sentence properties that come with the sentence structure. For example, when one wants to focus on a person who does an action, an active voice is used. On the other hand, when a person whom receives such action is focus, then a passive voice is used. Different view points can also demonstrate in the syntactic representation. Therefore, instead of choose one over another (either syntactic or semantic), combining both together can accommodate the drawbacks. The combined representation is called Syntactic-Semantic Graph (SSG). Its features are described in the next section. Note that SSG provides the semantic meaning of a sentence and, at the same time, it preserves the syntactic structure of such sentence.

Second, *recognizing paraphrases*: once each paraphrase pattern is defined, a recognition model is designed for the paraphrase pattern. These recognition models are explained in Chapter 6. The key point of recognition is to compare two sentence representations. The more they match, the closer the paraphrase is. There are cases in which a lesser match still counts or is considered as a paraphrase. Each of these cases should be explainable. This then becomes the third issue, *explicating paraphrase differences*. If different paraphrase patterns are found, each pattern should be explainable and demonstrate different impact on paraphrase recognition process. For example, if a pattern is a synonym, the paraphrase recognition process will be easier than changing part of speech or more than one patterns combined. The last issue (not implemented as part of this dissertation) is *measuring paraphrase distance*. When a paraphrase is recognized, all main points may not be covered; hence, a distance (or different) between the two sentences (or two sets of sentence) should be measured. A good paraphrase should be close to the original sentence while a fine paraphrase may not be; yet cover some main points.

CHAPTER 4

SENTENCE REPRESENTATION

The sentence representation used in this work is the *Syntactic-Semantic Graph* (SSG) which is based on *Conceptual Graph* (CG; Sowa 1983; 1992). The SSG is not a complete CG; but it has features of the semantics provided by CGs, while still keeping the sentence syntactic information which to be used in the disambiguation process. This chapter describes the SSG features and the SSG construction process.

Syntactic-Semantic Graph

A given sentence, whether it is an original or its attempted paraphrases, needs to be represented in some form for processing by the paraphrase recognition system. The SSG is a representation that includes both syntactic and semantic information, as the name suggests. Syntactic representation describes a sentence in grammatical terms: subject, verb, object, modifiers, tense, mode, and voice. Semantic representation contains conceptual relations among things or objects.

To represent a sentence in the syntactic portion of SSG, words are recognized and relations between words are tagged in syntactic-grammatical terms (*e.g.*, subject, verb, object, etc.) For example, "*A monkey eats a walnut*" consists of five words: *a, monkey, eat, a,* and *walnut* which are tagged as article, noun, verb, *etc.* The relation between *monkey* and *eat* is *subject*, between *eat* and *walnut* is *object*. The "a" is *determiner* of *monkey* and *walnut*. Another example, "*A walnut is eaten by a monkey*" consists of seven words. A relation between *walnut* and *eat* is *subject*, instead of *object*. However, it is a *subject* of a *passive voice* verb "*is eaten*." *Monkey* now connects to *by* as a modifier of a preposition. The summary of these two syntactic structures is shown in Figure 20 where *D* is a determiner, *S* is a subject, *O* is an object, *P* is for Passive voice, *MV* is a verb modifier (in this case prepositional phrase),and *J* is a modifier of a preposition.

```
"A monkey eats a walnut."

    +----------------Xp----------------+
    +-----Wd----+       +----Os---+    |
    |     +--Ds-+---Ss--+   +-Dsu-+    |
    |     |     |       |   |     |    |
LEFT-WALL a monkey.n eats.v a walnut.n .
```

```
"A walnut is eaten by a monkey."
    +---------------------Xp--------------------+
    +-----Wd----+                  +---Js--+    |
    |     +-Dsu-+--Ss--+--Pv-+-MVp-+ +--Ds-+    |
    |     |     |      |     |     | |     |    |
LEFT-WALL a walnut.n is.v eaten.v by a monkey.n .
```

Figure 20: Syntactic Structures by Link Grammar.

To represent a sentence in the semantic portion of an SSG, first concepts and relations are defined. A concept can be an object, thing, or action. A relation is the semantics of how one concept is related to another concept. For example, "*A monkey eats a walnut*" consists of three concepts: *Monkey*, *Eat*, and *Walnut*. A relation between *Monkey* and *Eat* is that *Monkey* is an *Agent* of *Eat*; whereas a relation between *Eat* and *Walnut* is that *Walnut* is a *Patient* of *Eat*. There are three notations: box, circle, and arrow. A box is used for a concept, a circle for a relation, and an arrow shows the direction of such relation. However, in linear text square brackets are used instead of boxes, and parentheses instead of circles. Arrows represent the direction of the relation so that the graph [CONCEPT_1] → (REL) → [CONCEPT_2] is read in English as "The REL of CONCEPT_1 is CONCEPT_2". Hence, from the above example, two sub-SSGs are written as follows:

[Eat] → (Agent) → [Monkey], which can be read, "The Agent of Eat is Monkey".

[Eat] → (Patient) → [Walnut], which can be read, "The Patient of Eat is Walnut".

And therefore, a SSG representing this sentence in linear text is:

[Monkey] ← (Agent) ← [Eat] → (Patient) → [Walnut]

SSG uses the individual as a concept. For example, in "Chomsky eats a walnut", the SSG will be [Chomsky] ← (Agent) ← [Eat] → (Patient) → [Walnut]. It does not matter whether *Chomsky* is a name of an individual monkey or person, SSG will refer to *Chomsky* as a concept.

A quantifier in SSG will be described as an *Article* or *Quantity* relation. For example, a SSG representing "a car" is [Car] → (Article) → [A], whereas representing "*five cars*" is [Car] → (Quantity) → [Five].

In addition to semantic relations, SSG includes syntactic relations. Note that these syntactic relations are in addition to what given by the Link Grammar parser. For example, a SSG representing "*John builds a house <u>with a hammer</u>*" is:

[Build] → (Agent) → [John]
[Build] → (Patient) → [House]
[Build] → (Verb_Prep) → [Hammer] {with}

A *Verb_Prep* relation is a syntactic relation describing that *Build* has a verb-preposition relation with *Hammer* through a preposition *with*. In linear text, curly brackets are used to indicate the preposition. Once the preposition *with* usage has been disambiguated (in this example, as an *instrument*), the SSG representing this sentence will be

[Build] → (Agent) → [John]
[Build] → (Patient) → [House]
[Build] → (Instrument) → [Hammer]

An *Instrument* relation is a semantic relation. Details of the preposition disambiguation are described in the subsequent Chapter.

Comparison to Conceptual Graph

As can be noticed in previous section that the SSG's basic features are similar to the *Conceptual Graph* (CG, Sowa 1983; 1992) and their differences are the kinds of relations included in the graph. That is SSG includes not only the semantic relations featured in the CG, but also the syntactic relations which are used in the disambiguation process. The following are comparison between SSG's and CG's features:

- Similar to CG, an SSG relation describes *a relation* between *two concepts*. Most relations used in SSG are *semantic* (also called *specific relations*), except some that are *syntactic* (also called *general relations*). For example, *Verb_Prep* is a syntactic relation indicating a general relation between a *verb* and a *preposition-modifier* prior to the preposition disambiguation process while after preposition disambiguation it could be transformed to a semantic relation, such as *Agent, Instrument, or Attribute* (as described in Chapter 4).
- There are some cases, unlike in CG, where SSG's relations are general. For example, CG has an *Accompaniment* relation to describe a relation of a *person* that *accompanies* an *agent* during an action. In SSG, the *Accompaniment* will be transformed to either an *Agent or Patient* relation based on the preposition classification result. This benefits the paraphrase recognition process. By disambiguating *Accompaniment*, there is no need for a special paraphrase rule that an *Accompaniment* relation could be interpreted as either *Agent or Patient.* If *Accompaniment* was not disambiguated, the paraphrase recognition module would require additional information to where this *Accompaniment* attached to (either attaches to an *Agent* or *Patient*). Hence, it adds more complexity in the recognition process. Note that SSG preserves a *subject* of a sentence; hence, the focus or emphasis on an *Agent* or *Patient* can be disclosed, similarly with an *object* of the sentence.

- SSG treats prepositions differently than CG. In some cases, CG treats prepositions as relations. For example, in one of Sowa's example sentences "*A cat is on the mat*", an *On* or *Loc* relation is used to represent the relation between *cat* and *mat*: [Cat] → (On) → [Mat] or [Cat] → (Loc) → [Mat]. In this case, the preposition disambiguation model of the SSG will distinguish among a variety of usages and choose the *location* relation in this example to construct the sentence's SSG.

- There are some cases, unlike CG that has general relations, SSG gives deeper semantic meanings in some relations. For example, *location* is a relation that is used in both CG and SSG, but it does not differentiate among possible locations. Is it on top of the surface, above the surface, or just near by that location? In SSG, *location_on_surface, location_above_surface,* and *location_point* are used to describe these deeper semantics for these locations. One important advantage in defining both shallow and deep meanings in the paraphrase recognition process is that, the shallow meaning would only consider *location* while a deeper meaning would take the entire relation identification (*e.g.*, *location_on_surface*). In this case, CG does not differentiate *location* relation, while SSG does.

- Unlike CGs where only the final semantic graph is generated, SSG preserves the original syntactic representation of a sentence, as shown in Figure 20. If needed, this information can be used.

In summary, for a given sentence, SSG is an intermediate meaning representation between syntactic and semantic representation. It mainly contains the semantic information; yet preserves the syntactic information. Consequently, no information is lost.

Syntactic-Semantic Graph Construction

To construct the Syntactic-Semantic Graph (SSG), different levels of processes are involved as shown in Figure 21. The first module is *Syntactic-Semantic Graph Generator* (SSGG), which constructs a SSG of a given natural language sentence using mapping rules. The result of this is the first level of SSG, which is close to syntactic level. Then, the first Level SSG is modified by the *Syntactic-Semantic Graph Refiner* (SSGR) in a process that utilizes knowledge-based refining rules to put more semantics into the representation. These rules include word sense disambiguation and basic graph transformation. The result of this SSGR is the second level SSG. Once the SSG is more to the semantic side, the *Syntactic-Semantic Graph Packer* (SSGP) concises the SSG in the case that sub-SSGs can be merged together. The result of this process is the third level SSG but, in many cases, the 2nd and 3rd level SSGs would be the same. So far, SSGG (described next) and SSGR (described in next Chapter) have been implemented.

The *Syntactic-Semantic Graph Generator* (SSGG) will generate a proper SSG based on the parse result, which then will be used in the preposition disambiguation and paraphrase recognition modules. The parse is created by the *Link Grammar*, developed at Carnegie Mellon University (2000), that assigns to a sentence a syntactic structure that consists of a set of labeled links connecting pairs of words. Valid word usage is represented by rules about how different words may be linked together. A valid sentence is one in which all words are connected with valid connecting rules (called *Link connectors*). Thus a parse is a solution to the problem of finding links to connect all the words in the sentence. The parser is able to skip some portions of the sentence that it cannot understand and is able to handle some unknown vocabulary. One major benefit from the Link Grammar parser is that it gives all possible solutions for a sentence. Since the paraphrase recognition takes an optimistic approach[12], these multiple parse results (or alternative parses) are examined and used to determine both whether the student is

[12] **The Optimistic approach** in this paraphrase recognition system examines all possibilities of paraphrases, whether or not it is a correct and/or complete paraphrase; whether or not there is a misunderstanding of the word meaning, and whether or not the connection or attachment between two words was correct. The optimistic approach will exercise all Link Grammar results and accept any of the result that provides the paraphrase.

attempting to paraphrase and whether the student has misunderstood a sentence.

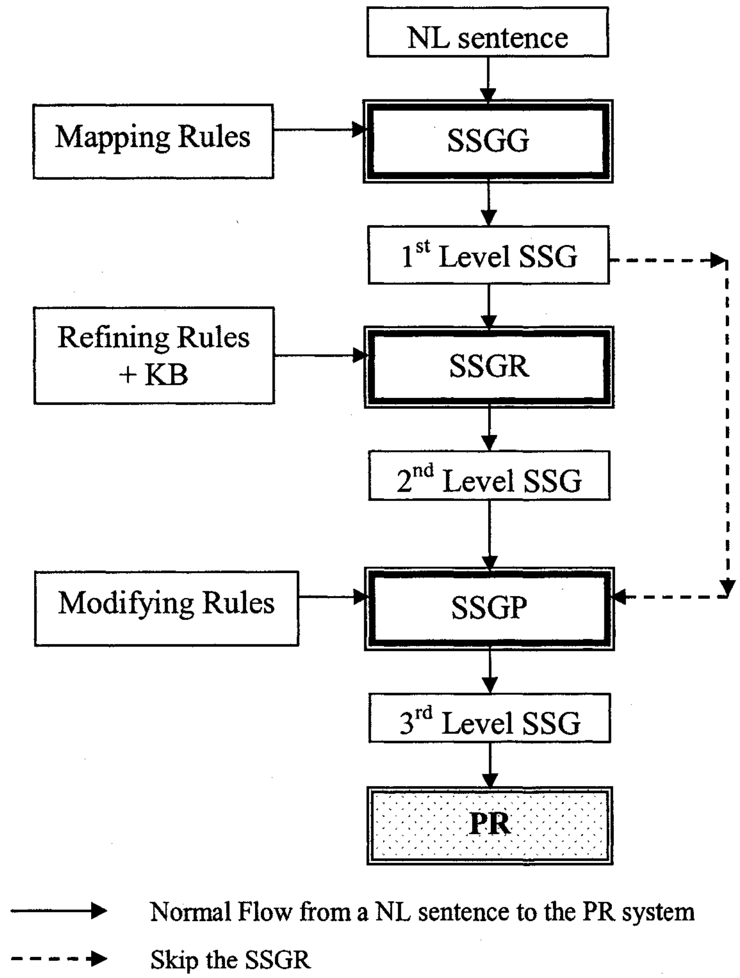


Figure 21: The System Architecture.

In addition, *WordNet* (Miller et al. 1990; Fellbaum 1998) is chosen as an ontology. As described in Chapter 2, words are connected in WordNet if they are related to one another. Currently the system uses synonym, hypernym, hyponym, antonym, meronym, and holonym relations. For a given word, WordNet can retrieve words related by all of these relationships. It can also produce words that have indirect relationships, such as the hypernym of a synonym. These word relations can be used in the analysis of potential paraphrases.

To construct a SSG of a sentence, for example, "*John builds a house with a hammer*", this sentence is first parsed by the Link Grammar. In this case, it produces two linkages as shown in Figure 22. The cost vector information shown (provided by Link Grammar) is not used in this system.

```
Linkage 1, cost vector = (UNUSED=0 DIS=0 AND=0 LEN=11)

    +----------------------Xp---------------------+
    |              +-------MVp------+             |
    |              +----Os---+      +---Js---+    |
    +---Wd--+--Ss--+    +-Ds-+      |  +--Ds-+    |
    |       |      |    |    |      |  |     |    |
LEFT-WALL John builds.v a house.n with a hammer.n .

Constituent tree:

(S (NP John)
   (VP builds
       (NP a house)
       (PP with
           (NP a hammer)))
   .)
```

```
Linkage 2, cost vector = (UNUSED=0 DIS=1 AND=0 LEN=9)

    +----------------------Xp---------------------+
    |              +----Os---+      +---Js---+    |
    +---Wd--+--Ss--+    +-Ds-+--Mp--+  +--Ds-+    |
    |       |      |    |    |      |  |     |    |
LEFT-WALL John builds.v a house.n with a hammer.n .

Constituent tree:

(S (NP John)
   (VP builds
       (NP (NP a house)
           (PP with
               (NP a hammer))))
   .)
```

Figure 22: Link Grammar's linkage results of a sentence "John builds a house with a hammer."

Words in the sentence are connected via *Link connectors*; each indicates the syntactic relation of that word to the sentence and to the linked word. These connections are represented in terms of triplets. Each triplet consists of (1) a starting word, (2) an ending word, and (3) a connector between these two words. From the above example, Linkage 1 parse triplets can be represented as shown in Figure 23.

```
[(0=LEFT-WALL)(1=John)(2=builds.v)(3=a)(4=house.n)(5=with)(6=a)(7=hammer.n)
 (8=.)]

[[0 8 (Xp)][0 1 (Wd)][1 2 (Ss)][2 5 (MVp)][2 4 (Os)][3 4 (Ds)][5 7 (Js)]
 [6 7 (Ds)]]
```

Figure 23: Linkage 1's triplets of a sentence "John builds a house with a hammer."

[1 2 (Ss)] means 'John' is the singular subject of 'build', *[2 5 (MVp)]* means 'build' is connected to the 'with' prepositional phrase and *[5 7 (Js)]* means the preposition 'with' has 'hammer' as its object. We then convert each Link triplet into a corresponding SSG triplet. The two words in the Link triplet are converted into two concepts of the SSG. To decide whether to put a word on the left or the right side of the SSG triplet, we define a mapping rule for each Link connector. For example, a Link triplet *[word-1 word-2 (S*)]* will be mapped to the 'Agent' relation, with *word-2* as the left-concept and *word-1* as the right-concept: *[Word-2] → (Agent) → [Word-1]*. The SSG triplets for this example sentence are shown in Figure 24.

```
0 [0 8 (Xp)]    -> #S#  -> - N/A -
1 [0 1 (Wd)]    -> #S#  -> - N/A -
2 [1 2 (Ss)]    -> #S#  -> [builds.v] -> (Agent) -> [John]
3 [2 5 (MVp)]   -> #M# MVp + J (6) #
                        -> [builds.v] -> (Verb_Prep) -> [hammer.n]
{with}
4 [2 4 (Os)]    -> #S#  -> [builds.v] -> (Patient) -> [house.n]
5 [3 4 (Ds)]    -> #S#  -> [house.n] -> (Article) -> [a]
6 [5 7 (Js)]    -> #S#  -> [with] -> (Prep_Object) -> [hammer.n]
7 [6 7 (Ds)]    -> #S#  -> [hammer.n] -> (Article) -> [a]
```

Figure 24: Linkage 1 SSG triplets of a sentence "John builds a house with a hammer."

Each line (numbered 0-7) shows a Link triplet and its corresponding SSG triplet. These will be used in the recognition process. The '*#S#*' and '*#M#*' indicate single and multiple mapping rules. Details of the mapping rules can be found in Appendix A. To validate these mapping rules, we manually generate expected SSGs of many sentences in the iSTART database using the rules for conceptual graph construction given by Sowa (1983, 1992, 2001). Then we can check the results of this automated Syntactic-Semantic Graph generator.

In summary, a natural language sentence is transformed into a Syntactic-Semantic Graph, which will be used in the paraphrase recognition system. The Link Grammar is used as a parser and mapping rules are used to construct the SSG. The next chapter will describe how to use this SSG in the paraphrase recognition system.

CHAPTER 5

PREPOSITION CLASSIFICATION

While constructing the sentence representation, it is clearly shown in Chapter 4 that in order to construct an appropriate representation, which would help the paraphrase recognition process, the preposition meaning or its usage in the sentence should be identified. This led to the work on preposition sense disambiguation. This chapter proceeds in two steps in the preposition classification project. The first one is to classify a single preposition, "with". This design demonstrates the advantages of using the features of the heads and complements of the preposition in the sentence as well as the word relations defined in WordNet. The second step generalizes this classification so that it can be applied to any preposition. In this work, I have applied it to the ten most frequently used prepositions based on the Brown corpus (Edict VLC, 2004).

Preposition Senses for "with"

The preposition "with" is used in a number of different ways, as shown in Table 1: Longman's Dictionary (Longman Group Ltd, 1995) gives 15 ways in which "with" can be used; Merriam-Webster online (Merriam-Webster, 1997) has 11; dictionary.com has 27 (Lexico, 1995); and LCS lists 5 senses (Dorr, 2001). The set of definitions in LCS' preposition lexicon is used to define the "with" senses. Because it is smaller and coarser compared to other dictionaries, LCS' preposition senses are more general: the 15 meanings of 'with' in Longman can be mapped to the five LCS senses. In the course of operationalizing these senses, it is helpful to distinguish different usages of each.

The following are the five "with" senses based on LCS positions (Dorr, 2001), which are each followed by the LCS definition:

Identification – to indicate a property or quality of an object.

```
(
:DEF_WORD "with"
:COMMENT "with: 1. (a) The book with the red cover"
:LANGUAGE English
:LCS (WITH Ident (Thing 2) (* Thing 9))
)
```

Possession – to indicate that someone or something has or possesses something.

```
(
:DEF_WORD "with"
:COMMENT "with: 1. (a) He left the book with the secretary;
                       He filled the cart with hay"
:LANGUAGE English
:LCS (WITH Poss (Event 2) (* Thing 12))
)
```

Collocation – to indicate that two or more objects or people are located in the same place.

```
(
:DEF_WORD "with"
:COMMENT "with: 1. (a) He went with Mary; He fought with Mary"
:LANGUAGE English
:LCS (CO Loc (nil 2) (* Thing 11))
)
```

Instrument – to indicate a tool used to complete an action or cause an action to occur.

```
(
:DEF_WORD "with"
:COMMENT "with: 1. (a) He stabbed the burglar with a knife"
:LANGUAGE English
:LCS (WITH Instr (nil 27) (* Thing 20))
)

(
:DEF_WORD "with"
:COMMENT "with: 1. (a) He covered the baby in blankets"[13]
:LANGUAGE English
:LCS (IN Instr (nil 27) (* Thing 20))
)
```

Intention – to indicate the feeling associated with or a reason for an action

```
(
:DEF_WORD "with"
:COMMENT "with: 1. (a) He hurried with his dinner"
:LANGUAGE English
:LCS (WITH Intent (nil 27) (* Thing 22))
)
```

[13] This example is as shown in LCS, but 'with' should be in place of 'in'.

In these categories a relation may go in either direction. For example, in the two example sentences of *Possession* above the complement of '*with*', that is secretary or hay, could be either a possessor or a possessee; and both cases are classified as *Possession.*

Disambiguation Model for "with"

To disambiguate a use of 'with', its head and complement are examined. The head is what comes prior to 'with', including nouns or noun phrases of the subject and/or object of the main verb as well as the verb or verb phrase. The complement is what comes after 'with' – nouns or noun phrases. Each of the nouns of the head and complement is categorized into one or more WordNet ontological categories. For example, a high level category like *physical object* (*e.g.*, house, hammer, secretary, cart, and hay are physical objects), or a more specialized category such as *person*, *container*, or *substance* (e.g. secretary is a person, cart is a container, and hay is a substance). Based on the categories of the head and complement, hypernym or meronym relationships between those categories, and the LCS descriptors of the verb, the meaning or possible meanings of "with" in the sentence is determined.

The following sections indicate how these elements are connected to each of the meanings of "with" and then present an algorithm for discovering the meaning of the preposition in a sentence.

Identification

"With" can be used to indicate a property of an object. In the phrase "the book *with* the red cover", the head of 'with' (book) has a relation *has-part* with its complement (cover) that is discovered through WordNet. This use of 'with' is called *IdentHasPart.*

Case	Features of Heads	Features of Complements
IdentHasPart	{physical object}	{physical object} + part-of *Head*
IdentPhysProp	{physical object}	{physical object} + property
IdentPersProp	{physical object} + person	{cognition, knowledge}

Table 2: The features of heads and complements distinguishing *identificational* usage of 'with.'

In "the telescope *with* a diameter of 100mm", the head and complement do not have the *has-part* relation. However, its complement (diameter) is a *property* of a *physical object*, used to describe its head (telescope), which is a *physical object*. This use of 'with' is called *IdentPhysProp*.

If the head is recognized as a *person*, as in "the man *with* experience," and the complement is a *property* of a *person*, in particular *cognition* or *knowledge* categories, the case is one of with *IdentPersProp*.

Table 2 shows the relationships between features of head and complement and those three senses of 'with'.

Possession

There are many ways to use 'with' to indicate one object possessing or being possessed by another. In the case of "Tom leaves a book *with* Mary" or "Tom leaves Mary *with* a book", both indicate that Mary has possession of a book. The difference between these two syntactic structures for the same use of 'with' is distinguished, so they are called *PossObj1* and *PossObj2*, respectively. "Tom leaves Mary *with* a book" could also mean Tom carries off a book leaving Mary behind, which is called *PossSubj*.

Case	Features of Heads	Features of Complements
PossObj1	{physical object} [syn: object] + verb-poss	{person}
PossObj2	{person} + verb-poss	{physical object} [syn: object]
PossSubj	{physical object} [syn: subject] + verb-poss	{person}
PossContSubs	{container} + verb-poss	{substance}
PossContObj	{container} + verb-poss	{physical object}

Table 3: The features of heads and complements distinguishing *possessional* usage of 'with.'

In "Tom filled the cart *with* hay", there is a different sense of possession. The head (cart) is a *container* that possesses a *substance* (hay; a complement of 'with'). This use of 'with' is then called *PossContSubs*.

To loosen the previous case, the complement can be any *physical object*, not necessarily *substance*; "Tom filled the car *with* people". This use of 'with' is called *PossContObj*.

In each of these cases, the verb is recognized as one possibly indicating possession from its LCS descriptors and the hypernym relations found in WordNet are used to categorize the nouns.

<u>Collocation</u>

'With' indicates a collocation of two persons in "Tom leaves John *with* Mary". Two possible collocation cases are "Tom and Mary together leave John" or "John and Mary are left together by Tom". These collocation uses of 'with' are called *CollocSubjPerson* and *CollocObjPerson*, respectively. If the objects are not people, as in "Tom puts a pencil *with* the book", this case is called *CollocObjs* since both are objects and *CollocSubjs,* when they are subjects.

Case	Features of Heads	Features of Complements
CollocSubjPerson	{person} [syn: subject]	{person}
CollocObjPerson	{person} [syn: object]	{person}
CollocObjs	{physical object} [syn: object]	{physical object} [syn: object]
CollocSubjs	{physical object} [syn: subject]	{physical object} [syn: subject]

Table 4: The features of heads and complements distinguishing *collocational* usage of 'with.'

Instrument

In "John builds a house *with* a hammer" and "she covers a baby *with* a blanket", 'with' indicates a use of an *instrument*. In the first example, the complement of 'with' (hammer) is ontologically an *instrument* in WordNet. This case is called *Instr.* In the second example, a blanket is not an *instrument*; yet is used as such based on the verb, which according to LCS requires or could have an instrument. This is called *InstrPhysObj*.

Case	Features of Heads	Features of Complements
Instr	verb-instr	{physical object} + instrumentality
InstrPhysObj	verb-instr	{physical object}

Table 5: The features of heads and complements distinguishing *instrumentional* usage of 'with.'

Intention

'With' can also be used to indicate the intention or manner, e.g. "John builds a house *with* passion", "her father faces life *with* a smile." The complement of 'with' for intention is in one of these WordNet categories: *feeling*, *act*, or *attitude*; this is called *IntenGen*.

Case	Features of Heads	Features of Complements
IntenGen		{feeling, act, attitude}

Table 6: The features of heads and complements distinguishing *intentional* usage of 'with.'

Some verbs sometimes require a specific preposition. To deal with those cases, based on the primitives of event/state and LCS entry information, verbs are classified into categories. For example, LCS entry for the verb '*fill*':

```
LCS (be ident (* thing 2)
           (at ident (thing 2)
              (fill+ed 9))
        (with poss (*head*)
           (* thing 16)))
```

Since the 'with' of the verb '*fill*' causes a possession, we called this verb a *verb-poss*. For example, "She fills a pail *with* water."

Another example on the verb '*change*':

```
:LCS (cause (* thing 1)
       (go ident (* thing 2)
           (toward ident (thing 2)
              (at ident (thing 2)
                   (change+ed 9))))
       ((* with 19) instr (*head*)
           (thing 20)))
```

In this case, 'with' of verb '*change*' requires an instrument; hence, we called this *verb-instr*. For example, "Mary changes her life style *with* the right diet." In addition to these, 'with' of any given verb can indicate identification (*verb-ident*), an intention (*verb-*

intent), or a collocation (*verb-colloc*).

In summary, the five categories for "with" senses are described as the initial work on disambiguating a preposition sense, that is, utilizing the features of the heads and the complements as well as applying the ontology defined in WordNet. The results of the "with" disambiguation are described in Chapter 8. In brief, the results are promising and lead to the work described in the next section: the general sense classification and the generalized sense disambiguation model.

Generalized Preposition Classification

In Chapter 8, the results of disambiguating the preposition "with" demonstrate the effectiveness of using the heads' and complements' features and WordNet ontology. In order to integrate the preposition disambiguation model into the paraphrase recognition, other prepositions (rather than just the preposition "with") have to be disambiguated. The sense-definition in the previous section is tightly specific to "with" and it might not cover other senses of other prepositions. Hence, a more general and broad-coverage sense classification is needed. This leads to the effort of work described in this section.

Prepositions are classified into seven general categories (or *coarse-categories*) based on preposition usages. Each preposition will then be given a set of *fine-categories* based on the features of the head and the complement. This is called preposition *case*. In this section, both coarse- and fine- categories will be described.

Seven Categories of Classification

These seven prepositions have been classified according to their usages – how prepositions are used and how they contribute to the meaning of a sentence – into the following seven general preposition categories: *Participant, Location, Time, Intention, Instrument, Identification,* and *Quantity.*

Participant – the preposition indicates that its head or complement participates in the event or action. This includes the following fine categories: agent, patient, accompaniment, object, recipient, beneficiary, experiencer, and object.

(4) a. "John builds the house *with* Mary." - Mary also participates in a 'build' action.
b. "John leaves the book *with* Mary." - Mary is a recipient of a 'leave' action.
c. "John gives the book *to* Mary." - Mary is a recipient of a 'give' action.
d. "John buys a present *for* Mary." - Mary is a beneficiary of a 'buy' action.

In (4.a), the preposition "with" is connected to a verb "build" and a person "Mary"; hence, it indicates a relation that "Mary" is participating in an action "build." Similarly in (4.b) and (4.c) that prepositions "with" connects to a verb "leave" and a person "Mary" and "to" connects to a verb "give" and a person "Mary", respectively, indicating a participation of "Mary" as a recipient of these two actions. In (4.d), the preposition "for" can be attached to either a verb "buy" or a noun "present" and a person "Mary." In the both cases, "for" indicates a relation that Mary is a beneficiary of the "buy" action or a recipient of the noun "present."

Location – the preposition indicates that its complement is the location where the event or action occurs. This includes direction, source, intermediate location, and destination.

(5) a. "John goes *to* New York." - New York indicates a destination.
b. "John moves *from* London." - London indicates a source location.
c. "Mary puts a book *on* the table." - A 'table' indicates a location.
d. "Mary drops a book *at* the library." - A 'library' indicates a destination.

In (5.a), the preposition "to" connects to a verb "go" and a city "New York" and it indicates that "New York" is a location destination of an action "go." Even without a verb, the "to" with a city still indicates a location destination, only the action is unknown. Similarly for the source location using the "from" in (5.b), it connects to a verb "move" and a city "London." The location includes places (*e.g.*, the library in (5.d)), tangible objects (*e.g.*, the table in (5.c)), and intangible objects (*e.g.*, the city border).

Time – the preposition indicates that its complement is temporally related to the event or action. This includes duration, specific date or day, time, and frequency.

(6) a. "John built the house *in* 7 days." – indicates a duration of 7 days

b. "John works *from* 9 a.m." – indicates a start time at 9 a.m.

c. "Classes start *at* 8 a.m. *on* Monday." – indicates a start time and day

d. "Mary stays *for* a week." – indicates a duration

In (6.a), the preposition "in" connects a verb "build" and a duration "7 days"; hence, it indicates a time duration of this action "build." Similar in (6.d), the preposition "for" connects a verb "stay" and a duration "a week", indicating a time duration of "stay." The cases in (6.b) and (6.c) are a preposition connecting to an action verb and a time that the action starts. That is, "9 a.m." is a start time of an action "work" (6.b) while "8 a.m." is a start time and "Monday" is a start day of an action "start." Prepositions can also indicate an end time, for example, (6.b) can be modified to "John works *from* 9 a.m. *to* 5 p.m.", which the preposition "to" indicates an end time of an action "work."

Intention – the preposition indicates that its complement is a purpose, a cause, a manner of the event or action.

(7) a. "John builds the house *with* passion." - indicates feeling toward action

b. "Mary plans *to* go to the Central Library." - indicates a goal for a 'go' action

c. "John died *of* cancer." - indicates a cause of death

d. "Mary studies hard *for* a better life in the future." - indicates a purpose

The preposition "with" in (7.a) connects a noun "passion" to a verb "build" indicating the feeling toward the "build" action. The preposition "to" in (7.b) connecting a verb "go" to a verb "plan" and the preposition "for" in (7.d) connecting a noun "life" to a verb "study" indicate a goal or a purpose of the action. In (7.c), the preposition "of" indicates a cause-effect relation between a verb "die" and a noun "cancer", which "of" are connected to.

Instrument – the preposition indicates that its complement is a tool used to complete the event or action. This also includes a use of materials, communication, and transportation.

(8) a. "John destroys the house *with* a hammer." - indicates a tool used
b. "John builds the house *with* marble." - indicates a material used
c. "John has torn his shirt *on* a nail." - indicates a tool used
d. "Mary travels *by* plane." - indicates a transportation

The preposition "with" in (8.a) connects a tool "hammer" to a verb "destroys" indicating a use of the tool in this action while "with" in (8.b) indicates a material used to build a "house" based on the noun "marble." Preposition "on" can be used to indicate a tool, as shown in (8.c). The preposition "by" in (8.d) indicates a medium or transportation used.

Identification – the preposition indicates that its head is a part or a property of its complement (and vice versa).

(9) a. "John builds the house *with* a kitchen." - indicates a kitchen is part of the house
b. "John buys the book *with* the red cover." - indicates a property of the book
c. "Mary is majoring *in* accounting." - indicates an identification
d. "John is an expert *on* dogs." - indicates an identification of an expert

To indicate identification, the prepositions mostly connect two (2) nouns, in which one can be a part of (*e.g.*, a kitchen and a house in (9.a), a property of (*e.g.*, a book and a cover in (9.b), and a characteristic of (*e.g.*, a major in accounting in (9.c) and an export on dogs in (9.d).

Quantity – the preposition indicates that its complement represents content, measure, and order.

(10) a. "John bought the car *for* \$500." - indicates the measure (cost) of the car
b. "A dinner is *at* a \$25 a plate." - indicates the measure (cost) of a dinner
c. "Mary bought a gallon *of* milk." - indicates a quantity of milk bought
d. "John runs *for* five miles." – indicates the measure (total distance) of running

The preposition can be used to indicate the quantity, amount, contain, or measure using "for" follows by a cost in (10.a) or "at" follows by a cost in (10.b), "for" follows by distance (10.d). The preposition "of" in (10.c) can be interpreted two ways: one as a quality of mile bought, which is "a gallon", another is an identification of a gallon in which milk is or was. However, this is not the preposition issue, but rather the sentence parser: whether to connect a verb "buy" to a "gallon" or to "milk."

These seven general categories are classified based on preposition's usage, similar to other preposition resources (*e.g.* grammar books, dictionaries, The Preposition Project). They cover the vast majority of prepositions' usage, namely, they can be mapped to/from other resources' usage classification. Thus, these general categories can appropriately be used.

The next section describes various scenarios that each preposition could be used to indicate relations between words or components, described above.

Usage-Cases

A preposition has a number of usage categories. To distinguish different preposition usages, a scenario is defined. For each preposition, a number of scenarios applied to each usage category[14] is defined. If a preposition can be used in different ways to the same usage category, then separate scenarios are defined. Each scenario (called a *usage-case* or *case*) consists of a preposition name, general category, usage-case identification, verb attribute, features of head/complement, relation between head and complement, syntactic role of head/complement, and a mapping rule to a Syntactic-Semantic Graph (SSG) relation (described in the next section). Table 7 provides examples of case definitions in a concise format. Details for other prepositions can be found in Appendix C.

The usage-cases are divided into two categories: if the features of head and complement can be clearly identified in the proposed preposition classification, this case is tagged as a *specific case*, and otherwise as a *general case*. Ontology categories used in

---

[14] Note that the ideas of using features of the head and complement are based on Alam's work (2004) in disambiguating preposition senses for the preposition "over" and using WordNet (Miller et al. 1990; Fellbaum 1998) ontological categories (although for different purposes) upon Harabagiu's idea (1996). The preposition definitions from Litkowski and Hargraves (2005; 2006) and Mohanty et al. (2004; 2005) were considered, but they are still under the development.

general cases are either the top-level ontology in WordNet (e.g., *entity, act, state*) or first-level children of the top-level ontology (*thing, physical object, location, sky* are first-level children of *entity*). The general case is used to cover broader and unclearly-identified categories defined in WordNet. The main reason of having two categories is to handle the unexpected categories as well as the unusual word-hierarchy defined by WordNet.

The SSG relations are defined for both *general* (or *shallow relation*) and *specific* (or *deep relation*) meanings to benefit the paraphrase recognition process. For example, *location* is a general meaning while *location_in, location_on_surface,* and *location_above_surface* are specific meanings that indicate different semantics of the location. Shallow meaning is not sufficient to distinguish prepositions in some senses; however, they can be used to define *close-paraphrase* (almost exact or almost a paraphrase). That is, it can be determined that two sentences talk about the same location, but position relative to the location is required for an exact paraphrase match. On the other hand, if we do not recognize this shallow meaning, then the only conclusion can be made here is that these two are different. The definition of relations can be found in Appendix B.

Prep	Category	Usage-Case	Head	Complement	HC Relation	SSG Mapping
With	Part	WithPartAgtAcmp	person (Subj)	person		Agent
With	Part	WithPartObjAcmp	person (Obj)	person		Patient
With	Loc	WithLoc_L		location		Location
With	Loc	WithLoc_A		area		Location
With	Inten	WithIntenGen_F		feeling		Manner
With	Inst	WithInstr		instrumentality		Instrument
With	Iden	WithIdentIsPart	physical_obj	physical_obj	Is-Part_Head	Has-Part
With	Qual	WithQualContSubs	container	substance		Location_In
With	Loc	WithLoc_(S)		space		Location_On_Surface
With	Inten	WithInten_(A)		act		Manner
With	Iden	WithIdent_(A)		attribute		Attribute
In	Loc	InLocAt_St		state		Location
In	Time	InTimeAt_S		season		Time
In	Time	InTimeAt_M		month		Time
In	Time	InTimeDur_U		time_unit		Time_Duration
In	Inst	InInstr_C		communication		Instrument
In	Inst	InInstrMatr_T		material		Instrument
In	Iden	InIdentIsPart	physical_obj	physical_obj	Is-Part_Comp	Is-Part
In	Part	InPart_(Rel)		relation		Patient
In	Loc	InLoc_(S)		space		Location_On_Surface
In	Inten	InInten_(M)		motivation		Manner
In	Quan	InQuan_(Q)		quantity		Quantity

Table 7: Usage-Case Definition. Examples of 'with' and 'in' case definition: specific and general (general usage-cases include parentheses).

Another benefit of using specific relations is to differentiate which meaning falls into the same general category such as *in* and *on*. For example[15], any two of "A house is on the hill", "A house is in the hill", "A house is to[16] the hill", "A house is at the hill", "A

[15] This is to illustrate different prepositions used in the same sentence structure. Hence, some sentences may seem artificial.
[16] This sentence using "to" is unrealistic, so this is only to demonstrate the use of a preposition in a location destination sense.

house is by the hill" are paraphrases based on a general relation *Verb_Prep* since they talk about the same location "the hill" of "a house". With deep relations (after the preposition disambiguation), the relations will differ: *location_in* is for a sentence with *in*, *location_on_surface* for *on*, *location_destination* for *to*, and just *location* for *at* and *by*.

In the case that fined-grain categories of prepositions are needed, these seven general categories can be extended by adding specialized usage-cases. For example, a *patient* in *Participant* category can be specialized into *beneficiary, experiencer* and *recipient*; a *Location_Under* into *location_under_surface* (below, touched) and *location_below* (below without touching).

Preposition Pairs

Some prepositions can be used interchangeably, that is, they are *synonyms* of one another. The following are examples of preposition pairs that can be used interchangeably, *i.e.*, *Preposition Synonym List*:

about ≈ on:	*A textbook about/on African History*
above ≈ over:	*The water came up above/over our knees.*
across ≈ over:	*The plane was flying over/across Denmark.*
over ≈ more than:	*You have to be over/more than 18 to see this film.*
by ≈ near:	*We live by/near the sea.* (By give a closer sense than near.)
during ≈ in:	*We'll be on holiday during/in August.*

Theoretically, these cases could be recognized without the parsing, disambiguation, and transformation. The simplest way is to substitute the preposition. The SSGs of these two sentences before preposition disambiguation would be the same; hence, an exact match in the recognition process. Then, the *prepositions* of both sentences have to be checked whether or not they could be used interchangeably. The current implementation of the paraphrase recognition system has not yet covered this interchangeable usage due to the time limitation.

Similarly, two prepositions can be paired based on their opposite meaning (as shown below), that is, a *Preposition Antonym List.*

over ≠ under:	*Mary walk over/under the bridge.*
above ≠ below:	*The airplane flies above/below the radar.*
out (of) ≠ in:	*Mary is out/in the office.*
on ≠ off:	*Mary put on / take off her dress.* (Different verbs are used)

Paraphrasing of Preposition Definition

Each usage of a preposition has a definition, that is, a phrase or a sentence describing a meaning of preposition and how it is used. For example, the definition of preposition "on" *location* usage is "above and touching a surface." Therefore, a paraphrase of "a book is on the desk" is "a book is placed above the desk and is touching the surface of the desk." To recognize the usage of preposition definition, the SSG of a definition first is generated. The triplet matching process will involve two sentence SSGs plus a definition SSG. The current implementation of paraphrase recognition does not cover the definition of a preposition. Yet, the definition of a preposition can be interpreted from the SSG relations as shown in examples below (details of other SSG relations are described in Appendix B).

"Location_On_Surface"

in [C1] → (*Location_On_Surface*) → [C2] indicates that [C1] is located *above* and *touches* a surface of [C2]. If two different situations have the same *Location_On_Surface* relation, then they both have something else other than a preposition to distinguish them. For example, (x, y) co-ordinates give an exact on-surface location of [C1] relative to [C2].

"*Location_Above_Surface*"

in [C1] → (*Location_Above_Surface*) → [C2] indicates that [C1] is located *above* a surface of [C2], but does not touch it. The height of [C1] from [C2] may be varied and that can be described by (x, y, z) co-ordinates; where *z* is the height measured from the surface. When two different situations have *Location_Above_Surface*, they distinguish one from another by additional information besides a preposition.

The focus of this research is on the paraphrase recognition; hence, the definitions of words (not only nouns or verbs, but also prepositions) are not explored in detail. As for prepositions, the disambiguation process will only use the relations describing the prepositions' meanings, discussed in detail in Chapter 7.

CHAPTER 6

PARAPHRASE RECOGNITION

To recognize paraphrasing, after converting natural language sentences into Syntactic-Semantic Graphs (described in Chapter 4), two SSGs are compared for matching according to paraphrasing patterns. The matching process is to find as many "concept-relation-concept triplet" matches as possible. A triplet match means that a triplet from the student's input matches with a triplet from the given sentence. In particular, the left-concept, right-concept, and relation of both sub-graphs have to be exactly the same, or the same under a transformation based on a relationship of synonymy (or other relation defined in WordNet), or the same because of idiomatic usage. It is also possible that several triplets of one sentence together match a single triplet of the other. At the end of this pattern matching, a summary result is provided: total paraphrasing matches, non-paraphrased information and additional information (not appearing in the given sentence).

Paraphrase Patterns and Recognition Model

In this section, I illustrate an approach to paraphrase pattern recognition on single sentences: using synonyms, changing the voice, changing part-of-speech, using a definition, and changing the sentence structure.

Preliminaries: Before starting the recognition process, two assumptions must be held: (1) all the information is at the sentence level: each sentence has various content words (excluding such 'stop words' as *a*, *an*, *the*, etc.); (2) each content word has a list of synonyms, antonyms, and other relations provided by WordNet (Fellbaum 1998). These relations can be accessed via the Java WordNet Library (JWNL, Didion 2004).

Single-Word Synonyms: First, when both SSGs have the same syntactic pattern, then a check is conducted to determine whether the words in the same position are synonyms, as shown in Figure 25.

```
"John helps Mary."
     [helps.v] --> (Agent) --> [John]
     [helps.v] --> (Patient) --> [Mary]

"John assists Mary."
     [assists.v] --> (Agent) --> [John]
     [assists.v] --> (Patient) --> [Mary]
```

Figure 25: Example of paraphrases using synonyms.

Voice: Even if the voice of a sentence is changed, it will have the same SSG. For example, SSG for both "John helps Mary" and "Mary is helped by John" are shown in Figure 26.

```
"Mary is helped by John."
     [helps.v] --> (Agent) --> [John]
     [helps.v] --> (Patient) --> [Mary]
```

Figure 26: Example of paraphrases by changing voices.

Though both graphs are the same, SSG preserved the sentence structure information which indicates that one SSG is a *passive voice* sentence. That is the *Pv* Link connector will be present in one of the Link triplets.

Part-of-speech: A paraphrase can be generated by changing the part-of-speech of some keywords. In the example shown in Figure 27, "help" is a verb in sentence 1, while it is a noun in sentence 2.

```
"John helps Mary."
     [helps.v] --> (Agent) --> [John]
     [helps.v] --> (Patient) --> [Mary]

"John gives help to Mary."  (or "John gives Mary help.")
     [gives.v] --> (Agent) --> [John]
     [gives.v] --> (Patient) --> [help.n]
     [gives.v] --> (Patient) --> [Mary]
```

Figure 27: Example of paraphrases by changing part-of-speech.

Definition: To recognize a use of definition, a word definition has to be generated into an SSG. For example, a "*history*" means "*the continuum of events occurring in succession leading from the past to the present and even into the future*" (from WordNet 2.0). An SSG for this definition is shown in Figure 28 and its simplified version (manually created) is shown in Figure 29.

```
[continuum] -> (Attribute) -> [Event]

[occur] -> (Patient) -> [Event]
[occur] -> (Manner) -> [Succession] {in}

[lead] -> (Initiator) -> [Succession]
[lead] -> (Source) -> [Time: Past] {from}
[lead] -> (Path) -> [Time: Present] {to}
[lead] -> (Path) -> [Time: Future] {into}
```

Figure 28: SSG of "history" definition.

```
[occur] -> (Patient) -> [Event]
[occur] -> (Manner) -> [Succession] {in}
[occur] -> (Source) -> [Time: Past] {from}
[occur] -> (Path) -> [Time: Present] {to}
[occur] -> (Path) -> [Time: Future] {into}
```

Figure 29: Simplified SSG of "history" definition.

From WordNet 2.0, the synonyms of '*past*', '*present*', and '*future*' are "*begin, start, beginning process*", "*middle, go though, middle process*", and "*end, last, ending process*", respectively. As shown in Figure 30, the use of 'begin', 'go-through', and 'end' are parts of the SSG of a "history" definition. Hence, these words are recognized as parts of or synonyms of words that are part of the "history" definition, and then it is a paraphrase.

```
"All thunderstorms have a similar life history."
     [thunderstorms.n] -> (Article) -> [all]
     [have.v] -> (Agent) -> [thunderstorms.n]
     [have.v] -> (Patient) -> [history.n]
     [history.n] -> (Article) -> [a]
     [history.n] -> (Attribute) -> [similar.a]
     [history.n] -> (Attribute) -> [life.n]

"Thunderstorms go through similar cycles.  They will begin the
same, go through the same things, and end the same way."
     [go.v] -> (Agent) -> [thunderstorms.n]
     [go.v] -> (Patient) -> [cycles.n] {through}
     [cycles.n] -> (Attribute) -> [similar.a]

     [begin.v] -> (Agent) -> [thunderstorms.n]
     [begin.v] -> (Patient) -> [same]

     [go.v] -> (Agent) -> [thunderstorms.n]
     [go.v] -> (Patient) -> [things.n] {through}
     [things.n] -> (Article) -> [same]

     [end.v] -> (Agent) -> [thunderstorms.n]
     [end.v] -> (Patient) -> [way.n]
     [way.n] -> (Article) -> [same]
```

Figure 30: Example of paraphrases using a definition.

Sentence Structure: The same thing can be said in a number of different ways. For example, "John builds a house *with* a hammer", can be paraphrased by "John *uses* a hammer *to* build a house", "John builds a house *by using* a hammer", "A house is built *by* John who *uses* a hammer", or "A house is built *by* John *using* a hammer." These sentences convey the same meaning, yet they have different syntactic structures and use different prepositions. In some cases, prepositions can be used interchangeably, for example "Mary covers the baby *with* blankets" vs. "Mary covers the baby *in* blankets." In many cases; however, changing prepositions means changing sentence structure, for example "There are sixteen ounces *for* every pound" vs. "Each pound consists *of* sixteen ounces." In addition to changing a sentence structure, an absence of a preposition can result in changing a part-of-speech, for example "a book *with* a green cover" vs. "a *green-covered* book". In this example, since 'with' indicates a property of a book, a property ('green cover') can change its part-of-speech from a *noun* to an *adjective.*

```
"John builds a house with a hammer."
     [builds.v] --> (Agent) --> [John]
     [builds.v] --> (Instrument) --> [hammer.n]
     [builds.v] --> (Patient) --> [house.n]
     [house.n] --> (Article) --> [a]
     [hammer.n] --> (Article) --> [a]

"John uses a hammer to build a house."
     [uses.v] --> (Agent) --> [John]
     [uses.v] --> (Manner) --> [build.v]
     [uses.v] --> (Patient) --> [hammer.n]
     [hammer.n] --> (Article) --> [a]
     [build.v] --> (Patient) --> [house.n]
     [house.n] --> (Article) --> [a]
```

Figure 31: Example of paraphrases changing sentence structures.

In accord with this example, there is a paraphrase recognition rule (shown in Figure 32) that is used during the paraphrase recognition process. A complete list of implemented paraphrase rules can be found in Appendix D.

```
[uses.v] -> (Manner) -> [VERB]
[uses.v] -> (Patient) -> [INSTRUMENT]

     ≅      [VERB] -> (Instrument) -> [INSTRUMENT]
```

Figure 32: An example of a paraphrase rule.

Paraphrase Recognition Rules

Paraphrase recognition rules are constructed systematically by looking at preposition usages from grammar books (Quirk et al., 1985; Swan, 1996) and dictionaries (*e.g.,* Longman, 1995; Merriam-Webster, 1997) to determine how individual prepositions are used and whether any pair can be used interchangeably. For different usages, different paraphrase rules are defined.

Each paraphrase rule consists of a name, a type, a pair of SSGs, and an additional relation (if required to indicate the relation between concepts in this paraphrase rule). The rule in Figure 32 can be put in the structure as shown in Figure 33, where the capitalized words are part-of-speech or category variables and non-capitalized words are required as indicated. From Figure 31, VERB.v would be "build"; AGENT.n, "John"; and INSTRUMENTALITY.n, "hammer".

```
ParaRuleDef (
   ParaRuleName: USE-TO-Do-Manner-Inst
   ParaType: Instrument
   LeftLink:  ( [VERB.v] -> (Agent) -> [AGENT.n];
                 [VERB.v] -> (Instrument) -> [INSTRUMENTALITY.n])
   RightLink: ( [VERB.v] -> (Agent) -> [AGENT.n];
                 [use.v] -> (Manner) -> [VERB.v];
                 [use.v] -> (Patient) -> [INSTRUMENTALITY.n])
)
```

Figure 33: Paraphrase rule structure and its sample.

Similarity Measure

The similarity between two sentences can be categorized into one of these four cases:

1. <u>Complete</u> paraphrase <u>without extra</u> information
2. <u>Complete</u> paraphrase <u>with extra</u> information
3. <u>Partial</u> paraphrase <u>without extra</u> information

4. Partial paraphrase with extra information

To distinguish between '*complete*' and '*partial*' paraphrasing, the triplet matching result is used. What counts as complete depends on the context in which the paraphrasing occurs. If a paraphrase is used as a *writing technique*, the 'complete' paraphrasing would mean that all triplets of the given sentence are matched to those in the student's input. If any triplets in the given sentence do not have a match, it means that the student is 'partially' paraphrasing at best. On the other hand, if a paraphrase is used as a *reading behavior or strategy*, the 'complete' paraphrasing may not need all triplets of the given sentence to be matched. Hence, this case only requires recognizing which part of the student's input is a paraphrase of a significant part of the given sentence. Consequently, the following questions have been raised: how to measure whether this student's input is an adequate paraphrase of a given sentence? Can information provided in the given sentence be used as a measurement? Namely, which parts of the given sentence are important? If so, how can it be used? An expert can answer some of these questions, especially on identifying essential elements of the given sentence.

My current research does not cover any automated similarity measurement, but rather I use a manual process to detect the sentence pair similarity. Namely, the results of each paraphrase pair (described in Chapter 8) were manually analyzed.

Implementation of Paraphrase Recognition

As mentioned at the beginning of this section, the paraphrase recognition is sometimes as simple as a matching the SSG triplets. My main contribution is to use not only single-exact matches (one-to-one triplet match), but also multiple matches (many-to-one or one-to-many triple match) as well as associated matches (using word relations such as synonym, antonym, hyponym, meronym). These matches are recognized via the paraphrase recognition rules defined in the Appendix D.

Figure 34 shows the pseudo algorithm of the paraphrase recognition. SSGlist_1 and SSGlist_2 are a list of SSG representations of sentence 1 and 2, respectively, constructed as described in Chapter 4.

```
For each SSG in SSGlist_1
   For each SSG in SSGlist_2
      For each triplet in SSG1
         For each triplet in SSG2
             Check if triplet1 and triplet2 are paraphrases
              [exact, synonym, antonym, hypernym, meronym]
                    If YES, mark that pair as covered and
                            put a pair in result list
         End for
      End for

      For those unmatched triplet in SSG1
          checkParaRule() // check paraphrase for more than
                          // a single simple match
                          // based on the paraphrase recognition rules
      End for
   End for
End for
```

Figure 34: Paraphrase recognition algorithm.

The number of possible paraphrase results is *O(m.n)*, where *m*, *n* are numbers of final SSGs for *S1* and *S2*, respectively. When *m* and *n* are very large, the number of possible paraphrase results become $O(n^2)$ or $O(m^2)$. That is, the more SSGs after the preposition disambiguation and SSG transformation, the more number of possible paraphrase results.

Figure 35 shows some results for the example shown in Figure 31. *S1* and *S2* are two sentences. For each, the *aFinalSSG* tag contains the ID of the sentence's SSG, which is followed by a listing of the SSG itself as a collection of triplets. For each paraphrase result, *SSG(X1, X2)* identifies the graphs being compared. The list of triplet matches *Triplet (T1, T2)* identifies which triplets from the two graphs are being compared. The result of the comparison is either an exact match (EXACT), match by synonym (SYNO), or a match by paraphrase rule.

```
S1: John helps Mary.
<aFinalCG ID="0">
--- Prep String: ORIGINAL 0
     2 [helps.v] --> (Agent) --> [John]
     3 [helps.v] --> (Patient) --> [Mary]

S2: John assists Mary.
<aFinalCG ID="0">
--- Prep String: ORIGINAL 0
     2 [assists.v] --> (Agent) --> [John]
     3 [assists.v] --> (Patient) --> [Mary]

Paraphrase S1 vs S2:
-- 1 -- CG(0,0)
   -- Triplet(2,2) SYNO
   -- Triplet(3,3) SYNO
```

```
S1: John builds a house with a hammer.
<aFinalCG ID="3">
--- Prep String: 5   - {0} Instrument
WithInstrPhysObj ##{Instrument}## [,hammer]  - build
     2 [builds.v] --> (Agent) --> [John]
     3 [builds.v] --> (Instrument) --> [hammer.n]
     4 [builds.v] --> (Patient) --> [house.n]
     5 [house.n] --> (Article) --> [a]
     7 [hammer.n] --> (Article) --> [a]

S2: John uses a hammer to build a house.
<aFinalCG ID="2">
--- Prep String: 1   - {1} Intention
ToInten_Verb ##{Manner}##  [,build]  - build
     2 [uses.v] --> (Agent) --> [John]
     3 [uses.v] --> (Manner) --> [build.v]
     4 [uses.v] --> (Patient) --> [hammer.n]
     5 [hammer.n] --> (Article) --> [a]
     6 [build.v] --> (Infinitive_Attr) --> [to]
     7 [build.v] --> (Patient) --> [house.n]
     8 [house.n] --> (Article) --> [a]

Paraphrase S1 vs S2:
-- 18 -- CG(3,2)
   -- Triplet(4,7) EXACT
   -- Triplet(5,8) EXACT
   -- Triplet(7,5) EXACT
   -- Triplet(LM:2+3,LM:2+3+4) USE-TO-Do-Manner-Inst
```

Figure 35: Example of paraphrase recognition result.

CHAPTER 7

PREPOSITION DISAMBIGUATION

Prepositions play a significant role in changing sentence structures of paraphrase patterns (as shown in Chapter 6) more than other paraphrase patterns. However, the significance of the preposition disambiguation process in the paraphrase recognition is yet to be explored. I started my work on disambiguation on the preposition "with". This explored the features of a model expanding on Alam's and Harabagiu's work. The promising results from disambiguating "with" led me to develop the generalized preposition disambiguation model. The preposition disambiguation process by itself is evaluated. Then, it is integrated into the paraphrase recognition system and the integrated system is evaluated. The last set of evaluation answers the question of "how much the preposition disambiguation improves the paraphrase recognition system?" The results are described in Chapter 8.

In this chapter, I describe the algorithm used to disambiguate or classify the preposition usage. The first part is for "with" and then the next part is the generalized preposition disambiguation model.

Disambiguation Algorithm for "with"

To disambiguate a meaning of 'with', the following steps are taken:

1. A sentence is parsed by Link Grammar and a SSG for the sentence is generated.
2. Within a parse produced by Link Grammar, a SSG triplet containing 'with' is selected.
3. For a selected SSG triplet, the head and complement of 'with' are identified and analyzed, using WordNet to determine the hypernyms of each and meronym relations between the two. Meronyms are also noted among the hypernyms of the head and complement and the number of levels involved is retained.
4. Possible senses may be determined from the complement's hypernyms in WordNet. If the hypernyms include any of the following:

a. Act, Feeling, or Attitude. The sense is *IntenGen*.

b. Person. The sense is *Colloc*. If hypernyms of the head include a Person and the head is semantically an agent (according to the SSG), then the sense is further classified as *CollocSubjPerson*. If the head is both a Person and semantically a patient, then the sense is *CollocObjPerson*.

c. Physical Object. If the head is also a Physical Object, then the sense is *Colloc*. If both complement and head are Agents, then the sense is *CollocSubjs*; if they are Patients, then the sense is *CollocObjs*.

d. Instrumentality. The sense is *Instr*. If the complement is a Physical Object and the head is a verb-instr, then the sense is *InstrPhysObj*.

e. Property. If the head is a Physical Object, then the sense is *IdentPhysProp*.

f. Cognition (person quality). If the head is a Person, then the sense is *IdentPersProp*.

5. The following cases are also checked:

   a. If the head is part of (meronym relation within 3 levels of hierarchy) the complement, or vice versa, then the sense is *IdentHasPart*.

   b. If the head is a Container (hypernym relation) and complement is a Substance, or vice versa, then the sense is *PossContSubs*. If the hypernym is not Substance, but still Physical Object, then the sense is *PossContObj*.

   c. If the head is a Person and the complement is a Physical Object, and the head verb is verb-poss, then it is *PossObj*. If the head is syntactically an object, then the sense is *PossObj1*; if the complement is an object, then the sense is *PossObj2*. If the head is a subject, then the sense is *PossSub*.

For example, let us consider the sentence "John builds a house *with* passion". One of the linkages from Link Grammar shows that the complement of 'with' is 'passion'. The hypernym tree of 'passion' is checked to see whether it is under 'feeling', 'act', and/or 'attitude' and if so, one of the results will indicate the *IntenGen* for this. A part of the output from the system for this example is shown in Figure 36. Notice that since ‘passion’ is also categorized under ‘cognition’, it meets the criteria for property of a person, *IdentPersProp*.

```
is identification
  - IdentPersProp_C ## Linkage# 0, Sense# 6, Tree# 0, Node# 1,
       Level# 0 [build,passion]   - build
  - IdentPersProp_C ## Linkage# 0, Sense# 7, Tree# 0, Node# 1,
       Level# 0 [build,passion]   - build
  - IdentPersProp_C ## Linkage# 1, Sense# 6, Tree# 0, Node# 1,
       Level# 0 [house,passion]   - build
  - IdentPersProp_C ## Linkage# 1, Sense# 7, Tree# 0, Node# 1,
       Level# 0 [house,passion]   - build

is intention
  - IntenGen_F ## Linkage# 0, Sense# 1, Tree# 0, Node# 1, Level# 0
       [build,passion]   - build
  - IntenGen_F ## Linkage# 0, Sense# 5, Tree# 0, Node# 1, Level# 0
       [build,passion]   - build
  - IntenGen_F ## Linkage# 1, Sense# 1, Tree# 0, Node# 1, Level# 0
       [house,passion]   - build
  - IntenGen_F ## Linkage# 1, Sense# 5, Tree# 0, Node# 1, Level# 0
       [house,passion]   - build
```

Figure 36: Example results from “passion” hypernym tree.

The results also show which Link Grammar linkage they are derived from, which sense of the complement of ‘with’, the tree of the hypernym relation, the node of the WordNet SynSet, and the hierarchy level (if there is a meronym relation). Words are used as they are with minimal stemming (only -s and -ed suffices were removed). However, during the paraphrase recognition process, which the present work is part of, different word forms will be considered.

At this point, the sense of nouns or verbs has not been determined, that is, all possible senses of a noun in WordNet and all possible classes of a verb in the LCS entry are examined. Disambiguating noun senses will be future work, as described in Chapter 10, that can be implemented using the existing approaches (described in Chapter 2). This will tell us whether disambiguating noun sense improves the performance of preposition disambiguation.

Generalized Disambiguation Algorithm Design

The results from "with" disambiguation were promising. That led to a further investigation with similar approach (using features of the heads and complements plus word ontology) but one that could be expanded to other prepositions. Therefore, the general preposition classifications are defined (as described in Chapter 5). In this section, the pseudo-algorithm for the generalized preposition disambiguation is described as shown in Figure 37.

```
1. A sentence is parsed by Link Grammar and a SSG for the sentence is
generated.
   (If Link Grammar produces more than one linkage,
       then that number of SSGs will be generated for
              that particular sentence.)

2. For each preposition found in the sentence and within each SSG,
      a SSG triplet containing the preposition (called target)
      is selected.

3. For a selected SSG triplet,
      the head and complement of the target preposition are identified.

4. For each usage-case of the target preposition (a row defined in Table 7),
      the head and complement are analysed using WordNet to determine
      the following cases:
       i.   the hypernym of each,
       ii.  the hypernym between the two
            (head is a kind of complement and vice versa)
       iii. the meronym relation between the two
            (head is a part of complement and vice versa)

5. If the criteria of that usage-case scenario are met,
     that scenario is selected as one of possible scenarios of this
     target preposition.
```

Figure 37: Generalized Preposition Classification Model.

For example, let us consider the sentence "*John builds a house with a hammer.*" The preposition in this sentence is 'with'. One of the linkages from Link Grammar shows that the complement of 'with' is 'hammer' and the head is 'house'. For each usage-case of 'with', the features (in Table 2) of head and complement are checked. For instance, in the 'WithInstr' case, only the complement is required and it should be an instrumentality category. In this case, 'hammer' is the kind of instrumentality; hence this 'WithInstr' usage-case is selected. From the current implementation, a portion of the output of this example is shown in Figure 38. This is a concise version of a result showing the main category, usage-case, head/complement, and the main verb of the sentence. With the optimistic approach, all possible usage-cases (or scenarios) will be checked and listed if they meet the criteria.

```
= S: John builds a house with a hammer. =
Preposition Senses:
     - Participant      WithPartObjs ##  [house,hammer]   - build
     - Participant      WithPart_(T) ##  [,hammer]   - build T=Thing
     - Instrument WithInstr ##  [,hammer]   - build
     - Intention  WithInten_(A) ##  [,hammer]   - build      A=Act
```

Figure 38: Results from preposition disambiguation process (concise format).

The detailed-format version of these results is shown in Figure 39 for the instrument sense. This version includes the associated Link Grammar, the sense of the complement of "with", the tree of the hypernym relation, the node of the WordNet SynSet, and the hierarchy level (if there is a meronym relation).

```
= S: John builds a house with a hammer. =
Preposition Senses:

is instrument
     - Instr ## Linkage# 0, Sense# 1, Tree# 0, Node# 3, Level# 0
        [build,hammer]   - build
     - Instr ## Linkage# 0, Sense# 1, Tree# 1, Node# 4, Level# 0
        [build,hammer]   - build
     - Instr ## Linkage# 0, Sense# 2, Tree# 0, Node# 3, Level# 0
        [build,hammer]   - build
     - Instr ## Linkage# 0, Sense# 2, Tree# 1, Node# 4, Level# 0
        [build,hammer]   - build
...
...
     - Instr ## Linkage# 1, Sense# 1, Tree# 0, Node# 3, Level# 0
        [house,hammer]   - build
     - Instr ## Linkage# 1, Sense# 1, Tree# 1, Node# 4, Level# 0
        [house,hammer]   - build
     - Instr ## Linkage# 1, Sense# 2, Tree# 0, Node# 3, Level# 0
        [house,hammer]   - build
     - Instr ## Linkage# 1, Sense# 2, Tree# 1, Node# 4, Level# 0
        [house,hammer]   - build
     - Instr ## Linkage# 1, Sense# 5, Tree# 0, Node# 3, Level# 0
        [house,hammer]   - build
     - Instr ## Linkage# 1, Sense# 5, Tree# 1, Node# 4, Level# 0
        [house,hammer]   - build
...
...
```

Figure 39: Results from preposition disambiguation process (detailed format).

SSG Transformation

After preposition disambiguation, each resultant usage-case will be transformed into the corresponding SSG relation. As shown in Table 7, for each usage-case, a mapping rule is defined to transform this case into a proper SSG relation, *e.g.*, a '*WithInstr*' case will be mapped to an '*Instrument*' relation and then the corresponding SSG triplet will be transformed by replacing *Verb_Prep* relation to *Instrument*, as shown in Figure 40.

```
- SSG triplet before disambiguation

      3 [2 5 (MVp)]    -> #M# MVp + J (6) #    ->
        [builds.v] -> (Verb_Prep) -> [hammer.n]   (with)

- One of disambiguated senses: WithInstr

       WithInstr : [Left-Concept] -> (Instrument) -> [Right-Concept]

- SSG triplet after disambiguation & transformation

      3 [2 5 (MVp)]    -> #M# MVp + J (6) #    ->
        [builds.v] -> (Instrument) -> [hammer.n]
```

Figure 40: SSG Transformation.

Note that after sentence representation construction, preposition disambiguation, and SSG transformation, each sentence will have a number of SSGs (not only from Link Grammar, but also from preposition disambiguation). The more preposition disambiguated resultants, the more final SSGs generated for that sentence. All of these SSGs will be used in paraphrase recognition for the optimistic approach. If the system requires a strict and concise result of paraphrases, then a content expert is required to identify the correct parsing of an original sentence, correct preposition usage category, and correct final SSG after the disambiguation process.

CHAPTER 8

RESULTS

This chapter contains results of implementations of the preposition disambiguation and paraphrase recognition processes proposed above.

The Results of "with" Preposition Disambiguation

The first preposition disambiguation is for the preposition "with." Fifteen (15) verbs were selected for the test-set corpus: *appoint, build, change, cover, decorate, drop, escape, examine, face, fill, force, give, greet, hold,* and *leave*. These verbs were selected using the following steps:

1. From the SUSANE corpus and texts used in the iSTART and RSAT projects (McNamara & Sinclair, 2004; Magliano & Millis, 2004), sentences containing "with" were selected.
2. From the selected sentences, approximately 30 distinct verbs were found that were used with "with".
3. After sorting the verbs alphabetically, the first 15 were selected. We plan to expand the corpus to cover all 30 verbs in the future.

For each verb, 8 sentences were selected from one of the existing corpora, online resources, or manually created sentences. Due to the scarcity of sentences that contain "with" and the limitation of the Link Grammar parser and the current implementation of the SSGG, it is necessary for us to manually construct or simplify some sentences to illustrate how the algorithm works.

Each of the 120 sentences was manually analyzed[17] to identify possible senses of "with."[18] Then, this evaluation was used to compare against the results obtained by this

---

[17] I, myself, manually analyzed these data with the guidance from my research supervisors.
[18] In the future, this analysis will be done independently.

system.

For each sentence, the 'with' sense result can be classified into one of the following:

1. *Exactly Correct*: the 'with' sense provided by this implementation is exactly the same as the expected sense.
2. *Partially Correct*: at least one of the resultant 'with' senses is the expected sense.
3. *Incorrect*: the result does not include the expected sense.
4. *No Link Result*: the Link Grammar does not include 'with' in its parse result.
5. *No Result*: there is at least one Link Grammar linkage, but the categories of head and/or complement existing in WordNet do not meet the criteria defined in Table 2 - Table 6.

A code has been assigned to each category above: 0, 1, -1, -99, and -55, respectively. With this set of implementation, the results are as following:

- 11 sentences are excluded since their results are either category 4 (No Link Result) or 5 (No Result).
- Out of the remaining 109 sentences, a total of 86 sentences contains at least one of the expected results:

a. 26 are category 1 (Exact Correct)

b. 60 are category 2 (Partial Correct)

This indicates 79% correctness of remaining 109 sentences and 72% overall.

- More than one 'with' senses were identified (result category 2) because

a. All possible senses of nouns and verbs are used resulting in different 'with' senses.

b. All 'with' senses are considered; hence, if any rules in Chapter 4 are met, then that sense is selected.

SID	Verb	Sentence	Head	Complement	With-Sense	My Result	Code
4	appoint	John appoints Bruce **with** Mary.	appoint + person	person	CollocSubjPers / CollocObjPers	CollocSubjPers / CollocObjPers	0
5	appoint	John appoints a manager **with** experience.	appoint + person	experience	IdentPersProp	IdentPersProp_C	0
9	build	John builds a house **with** a hammer.	build + house	hammer	Instr	Instr / InstrPhysObj / CollocSubjs / CollocObjs / IntenGen_A	1
10	build	John builds a house **with** a kitchen.	build + house	kitchen	IdentHasPart	IdentHasPart_R / InstrPhysObj / CollocSubjs / CollocObjs	1
11	build	John builds a house **with** Tom.	build + house	person	CollocSubjPers	CollocSubjPers	0
12	build	John builds a house **with** passion.	build + house	passion	IntenGen	IdentPersProp_C / IntenGen_F	1
13	build	John builds a web site **with** special equipment.	build + site	equipment	Instr	Instr / InstrPhysObj / CollocSubjs / CollocObjs	1
37	decorate	John decorates a tree **with** Christmas lights.	decorate + tree	lights	InstrPhysObj	IdentPhysProp / IdentPersProp_C / Instr / InstrPhysObj / CollocSubjPers / CollocObjPers / IntenGen_T	1
38	decorate	John decorates a tree **with** his parents.	decorate + tree	parents	CollocSubjPers	CollocSubjPers / CollocObjPers / InstrPhysObj	1
50	escape	Mary escapes to paradise **with** her friend.	escape + paradise	friend	CollocSubjPers	CollocSubjPers / InstrPhysObj	1
51	escape	The prisoner escaped his cell **with** his bare hands.	escape + cell	hand	InstrPhysObj	** No Link Result for with	-99
52	escape	John escaped from the prison **with** Tom	escape + prison	person	CollocSubjPers	CollocSubjPers	0
74	fill	I fill a large pail **with** water.	fill + pail	water	PossContSubs	InstrPhysObj / PossContSubs / PossContObj / CollocObjs	1
75	fill	A man filled a pail **with** a sieve.	fill + pail	sieve	Instr	Instr / InstrPhysObj / PossContSubs / PossContObj / CollocSubjs	1
98	greet	The French **greet** people with "Bon soir".	greet + people	phrase	InstrPhysObj	** No Result **	-55
99	greet	Mary greets people **with** the given instruction.	greet + people	instruction	InstrPhysObj	IntenGen_A	-1
100	greet	People usually greet **with** a hearty handshake.	greet	handshake	InstrPhysObj	** No Link Result for with	-99
113	leave	Tom leaves the house **with** Mary.	leave + house	person	CollocSubjPers / PossObj1	CollocSubjPers	0
115	leave	Tom leaves the book **with** the red cover.	leave + book	cover	IdentHasPart	IdentHasPart_R / Instr / InstrPhysObj / CollocSubjs / CollocObjs /IntenGen_A	1

Table 8: Sample Result. For each sentence, an expected with-sense is defined under column "With-Sense."

Table 8 shows sample sentences from the test set: expected "with" sense, result from the current implementation, and analysis of its correctness. The result from the system shown in "*My Result*" column, and "*Code*" column indicated the correctness (0=exactly correct, 1=partial correct, -1=incorrect, -99=No Link Result, -55=No Result from the current implementation)

The Results of Generalized Preposition Sense Disambiguation

In generalized preposition disambiguation, there are two cases of results: general cases vs. specific cases. The results from "with" preposition disambiguation are similar to the specific cases, since the features of head and complement are specific. As the name suggested, these rules defined and used in this generalized preposition are general and scaleable. Namely, if there are additional or specific cases to be detected, then a rule can be added without modifying any code. On the other hand, if there is a new sense, the program has to be modified.

To evaluate the system performance for each preposition (except "with" for which the procedure of sentence selection can be found in previous section), 120 sentences[19] were hand-selected from either Link Grammar sample sentences, one of the existing corpora, online resources, or were manually created. Due to the limitation of the Link Grammar parser and the current implementation of the Conceptual Graph generator, it is necessary to construct or simplify some sentences manually to illustrate how the algorithm works.

Each of the 120 sentences was manually analyzed by me (with guidance from the research supervisors) to identify possible senses[20]. Then, this evaluation was used to compare against the results obtained by this implementation.

[19] This existing sentence selection could bias the results, though I tried to be fair. The sentences were manually selected as described to serve as a preliminary test set. The sentences had to be parseable by the Link Grammar to be useable in the system. In future, independently created annotated-corpora would be used, such as iSTART protocols, SENSEVAL data set, or MSPC (Microsoft Paraphrase Corpus).
[20] In the future, this analysis will be done independently.

For each sentence, the sense result can be classified into one of the following categories:

1. *Exactly Correct*: the preposition specific usage-case provided by this implementation is exactly the same as the expected sense.
2. *Specific-case Partially Correct*: at least one of the resultant preposition specific usage-cases is the expected sense.
3. *General-case Partially Correct*: at least one of the resultant preposition general usage-cases is the expected sense, and it was not listed under the specific cases.
4. *Specific-case Incorrect*: the specific-case result does not include the expected usage category.
5. *General-case Incorrect*: the general-case result does not include the expected usage category.
6. *No Result*: there is at least one Link Grammar linkage, but the categories of head or complement or both existing in WordNet do not meet the criteria defined. This includes cases when prepositions are used as verb-particles or in idiomatic expressions.

The results are shown in Table 9. Overall, the precision of the generalized disambiguation model is 79% of sentences with resultant usage-cases, and 76% of all sentences.

Result	of	to	in	for	with	on	at	by	from	over
1	29	47	8	42	23	19	28	18	48	23
2	27	34	69	10	61	23	22	4	11	21
3	52	18	20	39	19	41	41	50	33	33
4	5	4	8	16	14	15	9	10	5	5
5	7	17	15	6	3	18	15	29	23	18
6	0	0	0	7	0	4	5	9	0	20
Total	120	120	120	120	120	120	120	120	120	120
A	108	99	97	91	103	83	91	72	92	77
B	120	120	120	113	120	116	115	111	120	100
C	0.90	0.83	0.81	0.81	0.86	0.72	0.79	0.65	0.77	0.77
D	0.90	0.83	0.81	0.76	0.86	0.69	0.76	0.60	0.77	0.64
E	0.47	0.68	0.64	0.46	0.70	0.36	0.43	0.20	0.49	0.44
F	0.43	0.15	0.17	0.35	0.16	0.35	0.36	0.45	0.28	0.33

A = No. of sentences contain correct senses
B = No. of sentences that could find preposition senses
C = Percent correctness of those that could find a result
D = Percent correctness of all 120 sentences
E = Percent correct from specific cases
F = Percent correct from general cases. [C = E + F]

Table 9: Preposition Sense Disambiguation Results. Ordered by the frequency of use in the Brown Corpus, this indicates the number of sentences that have been classified into each of the 6 categories.

The Results of Paraphrase Recognition System – Synthesized Corpus

The Synthesized Corpus consists of 192 sentence-pairs. For each pair, the first sentence is selected from the preposition disambiguation corpus and the 2nd sentence is a possible paraphrase (including incorrect paraphrase). Out of 192 pairs, 4 pairs were unable to be processed due to their length, leaving us with 188 pairs.

The current PR system produces more than one output and not in any preferred order for each sentence-pair, as described in Chapter 6. Hence, one of these outputs was chosen as a representative result for that pair, that is the highly like result and classified into one of the following categories:

1. *Correct Result or Complete Match*: a paraphrase sentence covers complete information in a given target sentence. That is, all SSG triplets are matched.
2. *Partial Match*: most of the SSG triplets are matched, but a few are unmatched. This is based on the ratio of a number of matched triplets and a number of unmatched triplets, between 50 – 70%. However, if the matched triplets are only articles or determiners, then this pair will not be tagged as a partial match.
3. *Incorrect Match*: the system could not match any of the SSG triplets including those inaccurate matching,
4. *Incorrect Result*: This includes all cases that the system does not handle (listed in Chapter 9).

Table 10 shows the results and a comparison between paraphrase recognition with preposition disambiguation and another one without disambiguation process. Adding preposition disambiguation to the paraphrase recognition system improves the recognition result either from incorrect to completely correct or partially correct, or from partially correct to completely correct in 21% of the 112 pairs (that the existing implementation of the paraphrase recognition successfully processes). The conclusion is that the system can recognize a paraphrase correctly 90% of the time.

Result	Without Preposition WSD	With Preposition WSD
1	36	49
2	48	52
3	28	11
4	76	76
Total 1	188	188
Total 2	112	112
A	84	101
B	75 %	90 %
C	25 %	10 %

Total 1 = total of all sentences
Total 2 = total of sentences that the system handle
A = No. of sentences contain correct senses (complete and partial, category 1 and 2)
B = Percent correctness of those that the system handles (A ÷ Total 2)
C = Percent incorrect

Table 10: Paraphrase Recognition Results. Comparison of the paraphrase systems: one with preposition disambiguation process and another one without preposition disambiguation process.

The Results of Paraphrase Recognition System – iSTART Corpus

The first iSTART dataset is from the SERT training conducted at Northern Illinois University (NIU). The self-explanations were collected from college students who were provided with SERT training and then tested with two texts, Thunderstorm and Coal. Both texts consisted of 20 sentences. The Thunderstorm text was self-explained by 36 students and the Coal text was self-explained by 38 students. The self-explanations were coded by an expert according to the following 4-point scale: 0 = vague or irrelevant; 1 = sentence-focused (restatement or paraphrase of the sentence); 2 = local-focused (includes concepts from immediately previous sentences); 3 = global-focused (using prior knowledge).

The steps of selecting the dataset used for this research are as follows:

1. Sentences with a maximum length of 15 words are selected. These sentences cover two topics: *Stages of Thunderstorm Development* and *The Origin of Coal.*

2. For each sentence, called a *target* sentence, a number of students' protocols are chosen based on the length (less than 20 words) and human judgments (*i.e.*, the expert at NIU).

   - A judgment of 1 indicates minimalist coverage. The system would expect to recognize this as a paraphrase, including a partial paraphrase.

   - A judgment of 2 indicates the coverage not only of the current sentence, but also of a nearby sentence.

   - A judgment of 3 indicates the use of outside information, such as the student's world knowledge or a sentence in the text that is not proximate; hence, the protocol may not contain any information from the current sentence. In addition, most of self-explanations which fall in this case are long and contain more words than the PR current system can handle. Therefore, this case is omitted.

The Origin of Coal[21] Seven target sentences were selected from this text and a total of 111 sentence-pairs were chosen and manually analyzed for their paraphrases and patterns. Of out these 111 pairs, 68 pairs contain the implication of paraphrases and/or contain more than one clauses. The system does not yet handle these cases because (1) in some cases implication is counted as elaboration strategy, (2) clauses involve pronoun resolution and ellipsis, which are also difficult problems to be solved. Hence, only 43 pairs were used and the PR system could correctly identify 84% as a correct paraphrase

[21] *Stages of Thunderstorm Development Three out of* seven target sentences were chosen, and 47 pairs were analyzed. Most of these 47 pairs are either the implication of paraphrases, clauses, or using definition. Only 3 pairs that can be handled by the current system and the system could detect all correctly; hence 100% accuracy.

Systems		Human (1, 2)	
		Kendall's	Pearson's
iSTART word matching	A. (0, 1, 2, 3)	0.213*	0.252**
	B. (1=0+1, 2= 2+3)	0.367**	0.310**
iSTART combined	A. (0, 1, 2, 3)	0.310**	0.349**
	B. (1=0+1, 2= 2+3)	0.454**	0.454**
PR system	C. (1, 2)	0.335**	0.335**

*. Correlation is significant at the 0.05 level (2-tailed).
**. Correlation is significant at the 0.01 level (2-tailed).

A. Using the original student levels computed in iSTART: 0=short, irrelevant, 1, 2, and 3.
B. Collapsing the student levels: 0 to 1 and 3 to 2.
C. Results of paraphrase recognition system, either 1 (partial/complete paraphrase) or 2.

Table 11: Paraphrase Recognition Results of iSTART dataset: correlation between systems.

Currently, the PR system is not yet automated. The step to identify a paraphrase of each sentence pair is as described in previous section. Namely, I manually checked triplet matches for partial and complete matches. If there is enough information for either partial or complete paraphrase, then 1 is assigned. Otherwise, a value of 2 is assigned. Further analysis is still needed to come up with a set of mathematical formula besides this simple condition, such as using Discriminant Analysis or Regression Analysis. This should improve the paraphrase recognition results.

Note that even though the human judgment score is 1, the current iSTART feedback system may not agree, i.e., it could give a value of 0, 2, or 3 for the quality of self-explanation. Therefore, the comparison between the iSTART evaluation itself and human judgments is used as a baseline. There are two portions of iSTART evaluation:

one only word matching (number of word matches in different benchmarks) and another one word matching with LSA cosine values (called, *iSTART combined system*; details can be found in McNamara et al., 2007). In addition, the iSTART score of 0 and 1 are combined as 1 (minimalist cases together) while score of 2 and 3 are combined into 2.

The results of the comparison to the human judgments are shown in Table 11. It is worth noting that the PR system is not as good as the combined system. This is because the combined system utilizes information beyond the current sentence, such as prior sentences, subsequence sentences, and the title of a text.

The second iSTART dataset is from explanations of sentences in "*The Origin of Coal*" text collected during iSTART training conducted at Old Dominion University. The selection is based on expert judgment with respect to paraphrasing.

The expert judged "Paraphrase Only" (Code=1) indicating that the explanations only contain a paraphrase. The system would be expected to recognize this as a paraphrase or partial paraphrase. In this case, the expert assessed how similar the explanation is to the target sentence: 1=similar to target sentence (resembles and has the same structure) and 2=distant to target sentence (changes of voice: active/passive, transformation: positive/negative, and changes in viewpoints). Also, a paraphrase is evaluated as to whether or not it is accurate: 0=inaccurate, 1=partially accurate and 2=accurate.

The expert judgment of "Paraphrase + Current Sentence Elaboration" (Code=6) indicates that the explanations may contain partial or complete paraphrases, but also additional information. The system would be expected to recognize this as a partial paraphrase (and in some cases a complete paraphrase) with additional information.

Selected 31 sentence-pairs were processed by the PR system in which produced, for each sentence-pair, a list of SSG pairs and matching results. These results were manually tagged as being a paraphrase (complete, partial, or none) and as having extra information or not. The correlations between PR results and human-judgment are shown in Table 12.

Systems		Human (1, 6)	
		Kendall's	Pearson's
iSTART word matching	A. (0, 1, 2, 3)	0.439*	0.429*
	B. (1=0+1, 2= 2+3)	0.508**	0.508**
iSTART combined	A. (0, 1, 2, 3)	0.530**	0.524**
	B. (1=0+1, 2= 2+3)	0.545**	0.545**
PR system	C. (1, 2, 5, 6, 7)	0.382*	0.505**
	D. (1, 2)	0.586**	0.586**

*. Correlation is significant at the 0.05 level (2-tailed).
**. Correlation is significant at the 0.01 level (2-tailed).

A. Using the original student levels computed in iSTART: 0=short, irrelevant, 1, 2, and 3.
B. Collapsing the student levels: 0 to 1 and 3 to 2.
C. Results of paraphrase recognition system: 1=complete paraphrase, 2=partial paraphrase, 5=no paraphrase but extra information (including implication and clauses), 6=complete paraphrase with extra information, and 7=partial paraphrase with extra information.
D. Results of paraphrase recognition system, either 1 (partial/complete paraphrase) or 2 with additional information.

Table 12: Paraphrase Recognition Results of iSTART dataset #2: correlation between systems.

The paraphrase recognition results from two iSTART datasets demonstrate that this PR module can be used efficiently to identify a paraphrase with/without extra information. The work remaining and not part of my dissertation is to integrate this paraphrase recognition module into the iSTART). In the next chapter, there are detailed analysis and a discussion of these results.

CHAPTER 9

ANALYSIS AND DISCUSSION OF RESULTS

This chapter discusses the results of the preposition disambiguation and of paraphrase recognition. This chapter also includes a list of issues that the current PR system cannot yet handle and that can be improved in the PR system in the future.

Analysis of the Results of Preposition Disambiguation

The results for the preposition 'over' are superior to Alam's work. Her work was only a design and evaluated the model manually. The result for the preposition 'of' is as good as Manhanty et al. (90% accuracy). The result for the preposition 'by' is low due to a usage of 'by' in the passive voice. The process of explicitly identifying whether a sentence is a passive voice sentence is not yet integrated in the disambiguation. Currently, the disambiguation model will use only available information based on Link Grammar and the features of the head and complements. Depending on the nouns and their ontology hierarchy, then different results were obtained. Overall the accuracy is about 80% and the model can be used for other prepositions besides these ten. In addition, these results are quite sufficient and effective for the paraphrase recognition system.

Sometimes results cannot be obtained or are incorrect due to limitations in one or more of the following components:

- *Link Grammar Parser* - its parse algorithm, words contained in its dictionary, and part-of-speech categories of words. Link Grammar has its own dictionary listing words in different files based on their part-of-speech. Then, if a word is not in Link Grammar dictionary, the system may guess or discard that word when parsing a sentence. Link Grammar uses its own syntactic rules to produce the parse results. These rules may not cover all possible sentences, especially those complex sentences. The SSG construction depends upon the parse resulting from the Link

Grammar. Hence, if the words do not exist or exist in a different part-of-speech category, then its parse algorithm will not produce accurate linkages for the SSG construction. Resulting in the incorrect result of SSG and consequently incorrect result for preposition disambiguation.

- *WordNet* - word senses, word categories (hypernym/meronym). WordNet has experts identify each of words, each senses, and how they are ontologically related. These structures are from these experts, which may not exhaustive and may not agree with all individuals. They, however, are commonly agreed upon by WordNet experts. The preposition classification process uses word senses and relations between words as defined in WordNet. Hence, if these classifications were not covered or misclassified, then the incorrect results were produced during the preposition disambiguation.
- *SSG Generator* – the mapping rules from Link Grammar to CG identify a target preposition and its head and complement. The mapping rules cover the majority of relations, but may not be exhaustive for some exceptional cases. Some of these special cases may result from the parse produced by the Link Grammar.
- *Human Analysis* - expected preposition usage. Experts could be influenced by or be biased when they identify the correct preposition classification. In some case, there may be a number of possibilities resulting in inaccuracy or misinterpretation.

Analysis of the Results of Paraphrase Recognition

From Table 10, it can be seen that even without the preposition disambiguation, the PR system is able to identify paraphrases with 75% accuracy. When preposition disambiguation is integrated, the result improves to 90%, which is 21% improvement. The result can still be improved, especially with respect to those that were classified into category 4 (*i.e.* incorrect results) due to one of the following reasons (which currently not implemented):

- *Definition.* A definition of a verb is used in one of the sentences, for example "*travel by car*" for "*drive*", "*travel by plane*" for "*fly*".
- *Idiomatic Expression.* An idiom is used in one of the sentences, for example "*Drinks are on Harold*" vs. "*Drinks are bought by Harold.*"
- *Special Pairs of Verbs.* Some pairs of verbs cause switching between Agent and Patient, for example "give and receive" in "*John <u>gives</u> a book to Mary*" vs. "*Marry <u>receives</u> a book from John.*"
- *Relationship among People.* Some pairs of nouns cause switching between Agent and Patient, for example "uncle and niece" in "*John is Mary's <u>uncle</u>*" vs. "*Mary is John's <u>niece</u>.*"
- *Special relation allowing Agent and Patient interchange.* Such as, career position - A noun indicating a career position can cause the switching between Agent and Patient, in an example "the President" in "*John is the <u>President</u> of the company*" vs. "*The <u>President</u> of the company is John.*" Another example is "Mathematics is the most important of the sciences" vs. "the most important of the sciences is mathematics." This case is not career position.
- *Compound-Verbs.* Multi-word verbs have special meaning, for example "is made of", "is made from", "is built from", "go over".
- *Part-of-Speech.* A word that can be used as a verb or a noun, but in the same situation the noun would require another verb to obtain a paraphrase, for example "*make change*" (noun) vs. "*change*" (verb).
- *Comparative Form.* The use of comparative forms in place of preposition, for example "*over*" vs. "*more than*" vs. "*-er*"
- *Subordinating Clauses.* The use of clauses or subordinating in a sentence involves additional implementation in generating a sentence representation so as to connect the clause to the right place with the right *relation*.

- *No relation between verbs in WordNet.* For example, "visit" vs. "go"; "run" vs. "end", "leave" vs. "give"; "concern" vs. "worry" vs. "caught attention"

From the results from iSTART dataset #1 (Table 11), the correlations between systems and human judgment of the PR system is as good as the iSTART word-matching system but a bit lower than the iSTART combined system (word matching and LSA). However, iSTART dataset #2 (Table 12) shows that the PR system can overcome both iSTART word matching and iSTART combined systems. One reason could be from the different human coding used in these sets. Set #1 uses 0, 1, 2, and 3, where 1 is minimalist and 2 is sentence focused locally. Hence, it does not specifically distinguish whether or to it is a paraphrase. Set #2, on the other hand, has a code for a paraphrase, *i.e.*, "paraphrase only". It is worth noting here that the PR does not take word count into consideration. Nor does it consider the words in previous sentences which are available to the iSTART system. Hence, the PR system can use these additional variables.

Overall, the current Paraphrase Recognition System can recognize most common paraphrase patterns, as shown in Chapter 8. Even though the system still does not handle a number of issues, some of these issues are difficult problems and are the subjects of ongoing research. Some solutions are mentioned in the conclusion chapter.

CHAPTER 10

CONCLUSIONS

The current paraphrase recognition model (without preposition disambiguation) produces acceptable results. Integrating it with preposition disambiguation improves the recognition success rate significantly. In addition the preposition disambiguation model itself shows significant results compared to existing related work. Nonetheless, there is much to be improved. The preposition disambiguation process can be improved by (i) disambiguating noun senses or using world knowledge or context information, (ii) ranking the disambiguated results for use in the paraphrase recognition process, and (iii) considering other factors besides heads and complements. The paraphrase recognition process can be improved by (i) handling cases of prepositions in metaphors and verb particles, relations between people and special noun relations (permitting a switch between Agent and Patient), multiple part-of-speech words, and comparative forms, (ii) giving an automated paraphrase recognition feedback – for example, if the student missed important information in their input, the system can respond "*It's good start, but you left out XXX, can you say something about it?*", (iii) utilizing an annotation of correct paraphrase by the expert – the expert will verify each sentence's paraphrases as well as identify significant part of the sentences (what would be counted as partial or completed) and this information can be used in the feedback system, and (iv) applying knowledge or information of surrounding text in the recognition process. Some of these issues are further explained in this section.

Currently all possible senses of nouns are given equal weighting. To narrow down choices for the sense of prepositions, a noun sense should be disambiguated. Hence, word sense disambiguation (WSD) should be added into the system. There are existing works on WSD as described in Chapter 2, which can be used in the preposition disambiguation purpose. The simplest approach is to apply some of the heuristic methods described in *e.g.*, Ciaramita and Altun (2006), Pedersen et al. (2005), Castillo et al. (2004), Purandare and Pedersen (2004), Nastase and Szpakowics (2001), Li (1995), Voorhees (1993). These methods include most frequently used senses and a default sense. Theoretically, disambiguating nouns will add more precision to the preposition sense disambiguation. Ultimately, it will improve the paraphrase recognition.

Even with the noun disambiguation or context information or both, the sense of a preposition may not be uniquely determined. To the best of my knowledge, the PWSD approaches described in Chapter 2 only provide one result. Namely, one preposition's usage classification is given as the output. That implies that when that result is wrong (or misclassified), there are no alternatives. To benefit from this disambiguation model in the paraphrase recognition process the system could consider for example the top 5 likely senses of a preposition. Further investigation is required to find the proper number for the top senses of a preposition to be considered. Then, the ranking of preposition senses is essential. Based on head and complement information, the most frequently used sense could be rated higher than ones which rarely occur. Similarly, the specific usage-cases should be rated higher than the general ones. This ranked result will also be presented to the user during the identification process of the appropriate sense.

There are two ways to use all plausible senses of prepositions. First, during the text preparation, these plausible senses (if ranking is implemented, the highest rank is presented first) are presented to an expert. This allows the expert to choose the correct sense, that is, how the sentence should be interpreted, how the preposition is being used in such sentence, and how words should be interpreted (if the WSD is integrated into the system). Second, during the paraphrase recognition, all plausible senses are matched against the sentence annotated by the expert and if at least one sense qualifies as a paraphrase, then that result is acceptable.

Prepositions are also used in metaphoric expressions, idioms (*e.g.*, *with it* – dressing in fashionable clothes; *with you* – understand someone's explanation; *over with* – completely finished), or verb particles (*e.g.*, *come up with*, *deal with*, *relate to*, *tie in*). Research on English verb particles (this is also part of *multiword expressions*) is on-going including Kim and Baldwin (2006), Cook and Stevenson (2006). The meaning of prepositions in this case is idiosyncratic and no general rule can be defined. Therefore, we currently are not considering the disambiguation of such uses.

Besides features of heads and complements, other information, such as context or knowledge from previous sentence(s), may be needed in the prepositional sense disambiguation; they are yet to be explored. Even so, this generalized disambiguation model has proved its adequacy to benefit the paraphrase recognition system.

There are a number of approaches to provide an automated recognition reporter. The simplest one is to use already existing *similarity measurements*, such as a simple word count comparison (McNarama et al., 2004), a cosine distance used in LSA (Landauer, Foltz, & Laham, 1998; McNarama et al., 2005), and a Kullback-Liebler distance (KL-distance, Steyvers & Griffiths, 2005; Boonthum et al., 2007) used in Topic Modeling.

Providing appropriate feedback and direct guidance to the students thorough the iSTART curriculum is the ultimate goal. The results provided by current PR system could be used to provide the final feedback. For example, if the number of triplet matches is high, covering a majority of both sentences, then the system knows that it is a paraphrase. If triplet matches are mostly EXACT, then that sentence is a repetition whereas if some matches are SYNONYM, then that sentence is considered a good paraphrase. If the sentence produced by a student contains unmatched triplets, these triplets could be matched with the previous sentence(s) in order to recognize a bridging (*i.e.* a paraphrase of a previous sentence). Another benefit to iSTART would be the sense disambiguation. In order to give precise feedback, the expert has to identify the correct sense of words (nouns, verbs, prepositions) and also the meaning of a sentence (based on Link Grammar parse results). If a student's explanation is found to be a paraphrase of a sense other than the one identified by the expert, then the feedback could tell the student that a word meaning was misunderstood or the sentence misinterpreted. More research is still needed to investigate the possibilities and possible solutions.

In summary, the featured preposition classification has introduced a new way to classify the sense of a preposition based on its usage (the relation between two things that it connects) rather than on its literal meaning. The models of preposition sense disambiguation and paraphrase recognition designed in this work have proved to be efficient (as shown in Chapter 8), and provide significant results, especially when the disambiguation is added to the paraphrase recognition system. Hence, this approach is proven to benefit paraphrase recognition systems, that is, question-answering systems and tutoring systems.

REFERENCES

Agirre, E., & Edmonds, P. (2006). Introduction. In Agirre and Edmons (eds.), *Word Sense Disambiguation: Algorithms and Application*, pp. 1-28. Springer.

Agirre, E., & Rigau, G. (1996). Word Sense Disambiguation using Conceptual Density. In *Proceedings of the 16th International Conference on Computational Linguistic (COLING)*, pp. 16-22, Copenhagen, Denmark.

Alam, Y. S. (2004). Decision Trees for Sense Disambiguation of Prepositions: Case of Over. In *Proceedings of the HLT-NAACL Computational Lexical Semantic Workshop*, pp. 52-59, Boston, MA, USA.

ASU Writing Center. (2000). Paraphrasing: Restating Ideas in Your Own Words. Available at http://uc.asu.edu/writing/paraphrasing.html.

BAC Writing Center. (2002). Paraphrasing. Available at http://www.the-bac.edu/writingcenter/wctipsheets/Paraphrasing/paraphrasing.html.

Bannard, C., & Baldwin, T. (2003). Distributional Models of Preposition Semantics. In *Proceedings of ACL-SIGSEM, Workshop on "The Linguistic Dimensions of Prepositions and their Use in Computational Linguistics Formalisms and Applications"*, pp. 169-80, Toulouse, France.

Barker, K. (1996). The Assessment of Semantic Cases Using English Positional, Prepositional and Adverbial Case Markers. Tech. Rep., TR-96-08, Department of Computer Science, University of Ottawa.

Barzilay, R., & Lee, L. (2003). Learning to Paraphrase: An Unsupervised Approach Using Multiple-Sequence Alignment. In *Proceedings of the Human Language Technologies and North American Association for Computational Linguistics (HLT-NAACL)*, pp. 16-23, Edmonton, Canada.

Boonthum, C. (2004a). *iSTART: Paraphrasing Recognition*. Ph.D. Dissertation Proposal, Department of Computer Science, Old Dominion University.

Boonthum, C. (2004b). iSTART: Paraphrase Recognition. In *Proceedings of the ACL Student Research Workshop*, pp. 31-36, Barcelona, Spain.

Boonthum, C., Levinstein, I. B., & McNamara, D. S. (2007) Evaluating Self-Explanations in iSTART: Word Matching, Latent Semantic Analysis, and Topic Models. In A. Kao & S. Poteet (Eds.), *Text Mining and Natural Language Processing*, pp. 91-106. Springer.

Boonthum, C., Toida, S., & Levinstein, I. B. (2005). Sense Disambiguation for Preposition 'with'. In *Proceedings of 2nd ACL-SIGSEM Workshop on "The Linguistic Dimensions of Prepositions and their Use in Computational Linguistic Formalisms and Applications*", pp. 153-162, Colchester, UK.

Boonthum, C., Toida, S., & Levinstein, I. B. (2006). Preposition Senses: Generalized Disambigutaion Model. In *Proceedings of the 7th International Conference on Computational Linguistics and Intelligent Text Processing* (CICLing), Lecture Notes in Computer Science, pp. 196-207. Springer Verlag.

British National Corpus Consortium. (2001). *The British National Corpus*, (version 2 BNC World). Available at http://www.natcorp.ox.ac.uk/.

Brna, P. (1999). *Prolog Programming A First Course: Knowledge Representation.* Computer Based Learning Unit. Available at http://computing.unn.ac.uk/staff/cgpb4/prologbook/.

Bruce, R., & Wiebe, J. (1994). Word sense disambiguation using decomposable models. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistic (ACL),* pp. 139-145, Las Cruces, NM, USA.

Burnard, L. (2000). *Reference Guide for the British National Corpus (World Edition).* The British National Corpus Consortium, Humanities Computing Unit, Oxford University Computing Services.

Cabezas, C., Resnik, P., & Stevens, J. (2001). Supervised Sense Tagging using Support Vector Machines. In *Proceedings of the 2nd International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL-2)*, pp. 59-62, Toulouse, France.

Carnegie Mellon University. (2000). Link Grammar, http://www.link.cs.cmu.edu/link/.

Castillo, M., Real, F., Atserias, J., & Rigau, G. (2004). The TALP Systems for Disambiguating WordNet Glosses. In *Proceeding of the 3rd Evaluation Exercises for Word Sense Disambiguation (SENSEVAL-3) Workshop*, pp. 93-96, Barcelona, Spain.

Cawsey, A. (1994). *Databases and Artificial Intelligence 3: Knowledge Representation and Inference.* Heriot-Watt University, Edinburgh, UK. Available At http://www.cee.hw.ac.uk/~alison/ai3notes/all.html.

Chan, Y. S., & Ng, H. T. (2005). Word Sense Disambiguation with Distribution Estimation. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1010-1015, Edinburgh, Scotland.

Cho, B., Michael, J., Rovick, A., & Evens, M. (2000). An Analysis of Multiple Tutoring Protocols. In *Proceedings of the Intelligent Tutoring Systems: 5th International Conference* (ITS), pp. 212-221, Montreal, Canada. Springer.

Ciaramita, M., & Altun, Y. (2006). Broad-Coverage Sense Disambiguation and Information Extraction with a Supersense Sequence Tagger. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (EMNLP), pp. 594-602, Sydney, Australia.

Cognitive Science Laboratory. (2001). *WordNet: A lexical database for the English language.* Princeton: NJ. Available at http://www.cogsci.princeton.edu/~wn/.

Connexor. (2002). *Connexor Machinese Syntax,* http://www.connexor.com/m_syntax.html.

Cook, P., & Stevenson, S. (2006). Classifying Particle Semantics in English Verb-Particle Constructions. In *Proceedings of the COLING/ACL workshop on Multiword Expression: Identifying and Exploiting Underlying Properties*, pp. 19-27, Sydney, Australia.

Costello, F. J., & Kelleher, J. D. (2006). Spatial Prepositions in Context: The Semantics of 'near' in the Presence of Distractor Objects. In *Proceedings of the 3rd ACL-SIGSEM, Workshop on Prepositions*, pp. 1-8, Trento, Italy.

Cowie, J., Guthrie, J., & Guthrie, L. (1992). Lexical disambiguation using simulated annealing. In *Proceedings of the 14th International Conference on Computational Linguistic, (COLING),* pp. 359-365, Nantes, France.

Cycorp. 2002. Cyc, http://www.cyc.com/.

Devitt, A., & Vogel, C. (2003). *Using WordNet hierarchies to pinpoint differences in related texts.* Tech Rep., Computational Linguistics Group, Department of Computer Science, Trinity College Dublin. TCD-CS-2003-27.

Didion, J. (2004). JWNL: Java WordNet Library. Available at http://sourceforge.net/projects/jwordnet.

Dorr, B. J. (2001). *LCS Verb Database, Online Software Database of Lexical Conceptual Structures and Documentation.* Available at http://www.umiacs.umd.edu/~bonnie/LCS_Database_Documentation.html.

Dorr, B. J., Hendler, J., Blanksteen, S., & Migdalof, B. (1995). Use of LCS and Discourse for Intelligent Tutoring: On Beyond Syntax. In Holland, M., Kaplan, J., & Sams, M. (eds.), *Intelligent Language Tutors: Balancing Theory and Technology*, pp. 288-309. Lawrence Erlbaum Associates, Hillsdale, NJ.

Edict VLC. (2004). *Word Frequency Lists,* http://www.edict.com.hk/textanalyser/wordlists.htm.

Edmonds, P., & Hirst, G. (2000). Reconciling fine-grained lexical knowledge and coarse-grained ontologies in the representation of near-synonyms. In *Proceedings of the Workshop on Semantic Approximation, Granularity, and Vagueness*, Breckenridge, CO, USA.

Edmonds, P., & Hirst, G. (2002). Near-synonymy and lexical choice. *Computational Linguistics*, 28(2), 105-144.

Fellbaum, C. (1998). *WordNet: an electronic lexical database.* The MIT Press: MA.

Francis, W. N., & Kucera, H. (1964). *Manual of Information to Accompany a Standard Corpus of Present-Day Edited American English, for use with digital computers.* Providence, R.I.: Department of Linguistics, Brown University.

Freedman, R., Zhou, Y., Glass, M., Kim, J., & Evens, M. (1998). Using Rule Induction to Assist in Rule Construction for a Natural-Language Based Intelligent Tutoring System. In *Proceeding of the Twentieth Annual Conference of the Cognitive Science Society*, pp. 362-367, Madison, WI, USA.

Glass, M. (2001). Processing Language Input in the CIRCSIM-Tutor Intelligent Tutoring System. In Moore, J.D., Redfield, C.L., & Johnson, W.L. (eds), *Artificial Intelligence in Education*, pp. 210-221. IOS Press.

Graesser, A. (2002). Introduction to the Psychology of Science Text Comprehension. In Otero, J., Leon, J.A., & Graesser, A.C. (eds.), *The Psychology of Science Text Comprehension*, pp. 1-15. Mahwah, NJ: Erlbaum.

Graesser, A., Person, N., & Hu, X. (2002). Improving Comprehension through Discourse Processing. In *Halpern, D.F., & Hakel, M.D.* (eds.), *New Directions for Teaching and Learning*, 2002 (Issue 89), 33-44. Jossey, UK.

Graesser, A., Person, N., Harter, D., & TRG. (2000). Teaching Tactics in AutoTutor. In *Modelling Human Teaching Tactics and Strategies: Workshop W1* (ITS), Montreal, Canada.

Graesser, A., Person, N., Harter, D., & TRG. (2001). Teaching Tactics and Dialog in AutoTutor. *International Journal of Artificial Intelligence in Education*, 12(3), 257-279.

Graesser, A., Wiemer-Hastings, K., & Wiemer-Hastings, P. (2001). Constructing Inferences and Relations during Text Comprehension. In Sanders, T., Schilperoord, J., & Spooren, W. (eds.), *Text Representation: Linguistic and Psycholinguistic Aspects*, pp. 249-271. Amsterdam/Philadelphia: Benjamins.

Greasser, A., Wiemer-Hastings, K., Wiemer-Hastings, P., Kreuz, R., & TRG. (1999). AutoTutor: A simulation of a human tutor. *Journal of Cognitive Systems Research*, 1, pp. 35-51.

Gruber, T. R. (1993). A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2), 199-220.

Harabagiu, S. M. (1996). An Application of WordNet to Prepositional Attachment. In *Proceedings of ACL-96 student session*, pp. 360-363, Santa Cruz (CA, USA).

Hartrumpf, S., Helbig, H., & Osswald, R. (2006). Semantic Interpretation of Prepositions for NLP Applications. In *Proceedings of the third ACL-SIGSEM, Workshop on Prepositions*, pp. 29-36, Trento, Italy.

Hawes, K. S. (2003). *Mastering Academic Writing: Write a Paraphrase Sentence.* ACAD 1100 Course Website, University of Memphis, Memphis: TN. Available at http://www.people.memphis.edu/~kshawes/mastwrit.html.

Hurst, G. (2003). Paraphrasing Paraphrased. Available at http://ftp.cs.toronto.edu/pub/gh/Hirst-IWP-talk.pdf.

Information Science Institute (ISI). (1993). Loom Knowledge Representation, http://www.isi.edu/isd/LOOM/.

Information Science Institute (ISI). (1997). PowerLoom Knowledge Representation, http://www.isi.edu/isd/LOOM/PowerLoom/.

Information Science Institute (ISI). (1998). HALogen Generator, http://www.isi.edu/licensed-sw/halogen/.

Inkpen, D., & Hirst, G. (2001). Building a lexical knowledge-base of near-synonym differences. In *Proceedings of the 2nd NAACL Workshop on WordNet and Other Lexical Resources,* pp. 47-52, Pittsburgh, PA, USA.

Inui, K., Fujita, A., Takahashi, T., Iida, R., & Iwakura, T. (2003). Text Simplification for Reading Assistance: A Project Note. In *Proceedings of the 2nd International Workshop on Paraphrasing: Paraphrase Acquisition and Applications (IWP)*, pp. 9-16, Sapporo, Japan.

Jackendoff, R. (1983). *Semantics and Cognition*. MIT Press, Cambridge: MA.

Jackendoff, R. (1990). *Semantics Structures*. MIT Press, Cambridge: MA.

Kahsima, K., Araki, K., & Tochinai, K. (2002). A Proposal of Paraphrasing Method Using Inductive Learning for Dialogue System. In *Proceedings of the 1st International Conference on Information Technology & Application (ICITA)*, pp. 169-17, Bathurst, Australia.

Kayaalp, M., Pedersen, T., & Bruce, R. (1997). A Statistical Decision Making Method: A Case Study on Prepositional Phrase Attachment. In Ellison, T.M. (Ed.) *CoNLL97: Computational Natural Language Learning*, pp. 33-42.

Kim, J., Freedman, R., & Evens, M. (1998). Responding to Unexpected Student Utterances in CIRCSIM-Tutor v.3: Analysis of Transcripts. In *Proceedings of the 11th International Florida Artificial Intelligence Research Symposium,* pp. 153-157, Sanibel Island.

Kim, J., Freedman, R., & Evens, M. (2000). Relationship between Tutorial Goals and Sentence Structure in a Corpus of Tutoring Transcripts. In *Proceedings of the Midwest Artificial Intelligence and Cognitive Science Society Conference*, pp. 124-131, Dayton, OH, USA.

Kim, S. N., & Baldwin, T. (2006). Automatic Identification of English Verb Particle Constructions using Linguistic Features. In *Proceedings of the 3rd ACL-SIGSEM, Workshop on Prepositions*, pp. 37-44, Trento, Italy.

Kirkpatrick, B. (1998). *Roget's Thesaurus of English Words and Phrases*. Harmondsworth, Middlesex, England: Peguin.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to Latent Semantic Analysis. *Discourse Processes*, 25, 259-284.

Langkilde, I. (2000). Forest-based Statistical Sentence Generation. In *Proceedings of the 1st North American Meeting of the Association for Computational Linguistics (NAACL)*, pp. 170-177, Seattle, WA, USA.

Langkilde, I. (2002). An Empirical Verification of Coverage and Correctness for a General-Purpose Sentence Generator. In *Proceedings of the International Natural Language Generation Conference (INLG)*, pp. 17-24, New York, NY, USA.

Lassen, T. (2006). An Ontology-based View on Prepositional Senses. In *Proceedings of the third ACL-SIGSEM, Workshop on Prepositions*, pp. 45-50, Trento, Italy.

Leacock, C., & Chodorow, M. (2003). C-rater: Automated scoring of short-answer questions. *Computers and the Humanities*, 37(4), 389-405.

Leacock, C. (2004). Statistical Analysis of Text in Educational Measurement. In *Proceedings of the 7th International Conference on the Textual Data Statistical Analysis (JADT)*, pp.35-41, Louvain La Neuve, Belgium.

Lee, Y. K., Ng, H. T., & Chia, T. K. (2004). Supervised Word Sense Disambiguation with Support Vector Machines and Multiple Knowledge Sources. In *Proceedings of the 3rd International Workshop on Evaluating Word Sense Disambiguation Systems (SENEVAL-3)*, pp. 137-140, Barcelona, Spain.

Leech, G. (2000). *A Brief Users' Guide To The Grammartical Tagging of The British National Corpus*. Available at http://www.natcorp.ox.ac.uk/docs/gramtag.html.

Lesk, M. (1986). Automated sense disambiguation using machine-readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the 1986 SIGDOC Conference, Association for Computing Machinery,* pp.24-26, Toronto, Canada.

Levin, B. (1993). *English Verb Classes and Alternations: A Preliminary Investigation*, University of Chicago Press, Chicago: IL.

Lexico Publishing Group, LLC. (1995). Dictionary.com.

Li, X., Szpakowicz, S., & Matwin, S. (1995). A WordNet-based Algorithm for Word Sense Disambiguation. In *Proceedings of International Joint Conferences in Artificial Intelligence (IJCAI)*, pp. 1368-1374, Montreal, Canada.

Lin, D., & Pantel, P. (2001a). DIRT – Discovery of Inference Rules from Text. In *Proceedings of the 7th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 323-328. ACM Press, New York, NY.

Lin, D., & Pantel, P. (2001b). Discovery of Inference Rules for Question Answering. *Natural Language Engineering* 7(4), 343-360.

Lin, D. (2003). Dependency-Based Evaluation of Minipar. In Abeille, A. (ed.), *Building and using Parsed Corpora*, Dordrecht: Kluwer.

Litkowski, K. & Hargraves, O. (2005). The Preposition Project. In *Proceedings of the 2nd ACL-SIGSEM, Workshop on "The Linguistic Dimensions of Prepositions and their Use in Computational Linguistic Formalisms and Applications"*, pp. 171-179, Colchester, UK.

Litkowski, K., & Hargraves, O. (2006). Coverage and Inheritance in The Preposition Project. In *Proceedings of the third ACL-SIGSEM, Workshop on Prepositions*, pp. 37-44, Trento, Italy.

Litkowski, K. (2002). Digraph Analysis of Dictionary Preposition Definition. In *Proceedings of the ACL-SIGLEX, SENSEVAL Workshop on "Word Sense Disambiguation: Recent Success and Future Directions"*, pp. 9-16, Philadelphia, PA, USA.

Longman Group Ltd. (1995). *Longman Dictionary of Contemporary English* (3rd Edition). Longman, Harlow: UK.

Longman Group Ltd. (2005). *Longman Dictionary of Contemporary English* (4th Edition). Longman, Harlow: UK.

Luger, G. (2002). *Artificial Intelligence: Structures and Strategies for Complex Problem Solving* (4th Edition). Pearson Education: Addison-Wesley.

Magliano, J., & Millis, K. K. (2004). *RSAT texts.* Tech. Rep., Psychology Department, Northern Illinois University.

Magnini, B., Strapparava, C., Pezzulo, G., & Gliozzo, A. (2002). The Role of Domain Information in Word Sense Disambiguation. *Natural Language engineering,* 8(4), 359-373.

Mann, B. (1999). An Introduction to Rhetorical Structure Theory. Available at http://www.sil.org/~mannb/rst/rintro99.htm.

Mann, W., & Thompson, S. (2000). Two Views of Rhetorical Structure Theory. In *Proceedings of the Society for Text and Discourse Conference*, Lyon (France).

Marshall, D. (2000). Artificial Intelligence II: courseware. Available at http://www.cs.cf.ac.uk/Dave/AI2/AI_notes.html.

McNamara, D. S. & Scott, J. L. (1999). Training reading strategies. In *Proceedings of the Twenty first Annual Meeting of the Cognitive Science Society.* Hillsdale, NJ: Erlbaum.

McNamara, D. S., & Sinclair, G. (2004). *iSTART texts.* Tech. Rep., Department of Psychology, The University of Memphis.

McNamara, D. S. (2004). SERT: Self-Explanation Reading Training. *Discourse Processes*, 38(1), 1-30.

McNamara, D. S., Boonthum, C., Levinstein, I. B., & Millis, K. K. (2007). Using LSA and word-based measures to assess self-explanations in iSTART. In T. Landauer, D.S. McNamara, S. Dennis, & W. Kintsch (Eds.), *LSA: A Road to Meaning*, Mahwah, NJ: Erlbaum.

McNamara, D. S., Levinstein, I. B., & Boonthum, C. (2004). iSTART: Interactive strategy training for active reading and thinking. *Behavior Research Methods, Instruments, & Computers*, 36(2), 222-233.

Merlo, P., & Leybold, M. (2001). Automatic Distinction of Arguments and Modifiers: the Case of Preposition Phrases. In *Proceeding of the ACL workshop on Computational Natural Language Learning (CONLL)*, pp. 1-8, Toulouse, France.

Merriam-Webster, Inc. (1997). Merriam-Webster Online, http://www.m-w.com/.

Mihalcae, R., & Moldovan, D. I. (1999). A method for word sense disambiguation of unrestricted text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistic (ACL),* pp. 152-158, MD, USA.

Miller, G. (1995). WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11), 39-41.

Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., & Miller, K. J. (1990). Introduction to WordNet: An On-line Lexical Database. *International Journal of Lexicography*, 3(4), 235-312.

Mitchell, B. (2004). Towards More Accurate PP Attachment even with Simple Algorithms. In *Proceedings of 2nd ACL-SIGSEM Workshop on "The Linguistic Dimensions of Prepositions and their Use in Computational Linguistic Formalisms and Applications"*, United Kingdom, pages 110–118, Colchester, UK.

Mohanty, R. K., Almeida, A. F., & Bhattacharyya, P. (2005). Prepositional Phrase Attachment and Interlingua. In Cardenosa, J., Gelbukh, A., & Tovar, E. (eds), *UNIVERSAL NETWORKING LANGUAGE: Advances in Theory and Applications*. Special issue of *Research on Computing Science*, Instituto Politecnico Nacional, Mexico.

Mohanty, R. K., Almeida, A. F., Samala, S., & Bhattacharyya, P. (2004). The Complexity of OF in English. In *Proceedings of the International Conference on Natural Language Processing (ICON)*, Hyderabad, India.

Molla, D., Schwitter, R., Rinaldi, F., Dowdall, J., & Hess, M. (2003). ExtrAns: Extracting Answers from Technical Texts. *IEEE Intelligent System* 18(4), 12-17.

Montes-y-Gómez, M., Gelbukh, A., & López-López, A. (2000). Comparison of Conceptual Graphs. In Cairo, O., Sucar, L.E., & Cantu, F.J. (eds.), *MICAI 2000: Advances in Artificial Intelligence. Lecture Notes in Artificial Intelligence N 1793*, pp. 548-556. Springer-Verlag.

Montes-y-Gómez, M., Gelbukh, A., & López-López, A. (2002). Detecting deviations in text collections: An approach using conceptual graphs. In Coello, C.A., Albornoz, A.D., Sucar, L.E., & Battistutti, O.C. (eds.), *MICAI-2002: Mexican International Conference on Artificial Intelligence. Lecture Notes in Artificial Intelligence N 2313*, pp. 176-184. Springer-Verlag.

Montes-y-Gómez, M., Gelbukh, A., López-López, A., & Baeza-Yates, R. (2001). Flexible Comparison of Conceptual Graphs. *Lecture Notes in Computer Science 2113*. Springer-Verlag.

Murata, M., & Isahara, H. (2001). Universal Model for Paraphrasing – Using Transformation Based on a Defined Criteria. In *Proceedings of the 6th Natural Language Processing Pacific Rim Symposium (NLPRS), Workshop on Automatic Paraphrasing: Theories and Application*, pp. 299-306, Tokyo, Japan.

Nastase, V., & Szpakowics, S. (2001). Word Sense Disambiguation in Roget's Thesaurus Using WordNet. In *Proceedings of NAACL, WordNet&Other Lexical Resources Workshop*, pp. 17-22, Pittsburgh, PA, USA.

Nicholas, N. (1995). Parameters for an Ontology of Rhetorical Structure Theory. *University of Melbourne Working Papers in Linguistics,* 15, 77-93.

O'Hara, T., & Wiebe, J. (2002). Classifying Preposition Semantic Roles using Class-based Lexical Associations. Tech Rep., New Mexico State University. TR NMSU-CS-2002-013.

O'Hara, T., & Wiebe, J. (2003). Preposition Semantic Classification via Penn Treebank and Framenet. In *Proceedings of the Conference on Natural Language Learning (CoNLL)*, pp. 79-86, Edmonton, Canada.

Paul, D. B., & Baker, J. M. (1992). The design for the Wall Street Journal-based CSR corpus. In *Speech and Natural Language Workshop*, Morgan Kaufmann.

Pedersen, T., Banerjee, S., & Patwardhan, S. (2005). Maximizing Semantic Related to Perform Word Sense Disambiguation. *Research Report UMS1 2005/25.*

Person, N., Graesser, A., & TRG. (2000). Designing AutoTutor to be an Effective Conversational Partner. In *Proceedings of the Fourth International Conference of the Learning Sciences*, pp. 246-253. Mahwah, NJ: Erlbaum.

Purandare, A., & Pedersen, T. (2004). Unsupervised Word Sense Discrimination by Clustering Similar Contexts. *Research Report UMS1 2004/146*, August 2004.

Qiu, L., Kan, M., & Chua, T. (2006). Paraphrase Recognition via Dissimilarity Significance Classification. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing* (EMNLP), pp. 18-26, Sydney, Australia.

Quality Writing Center (QWC). (2002). Paraphrasing and Summarizing. Available at http://www.uark.edu/campus-resources/qwrtcntr/resources/handouts/parasum.html.

Quirk, R., Greenbaum, S., Leech, G., & Svartvik, J. (1985). *A comprehensive grammar of the English language.* Longman: London.

Rigau, G., Agirre, E., & Atserias, J. (1997). Combining unsupervised lexical knowledge methods for word sense disambiguation. In *Proceedings of joint 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics (ACL/EACL)*, Madrid, Spain.

Rinaldi, F., Dowdall, J., Kaljurand, K., Hess, M., & Molla, D. (2003). Exploiting Paraphrases in Question Answering System. In *Proceedings of the ACL: Workshop in Paraphrasing*, pp. 25-32, Sapporo, Japan.

Roget, P. (1852). *Roget's Thesaurus of English Words and Phrases.* Harlow, Essex, England: Longman Group Ltd.

Saint-Dizier, P., & Vazquez, G. (2001). A Compositional Framework for Prepositions. In *Proceedings of ACL-SIGSEM, International Workshop on Computational Semantic*, Tilburg, Netherlands.

Schank, R. C., &. Abelson, R. P. (1977). *Scripts, Plans, Goals and Understanding.* Hillsdale: New Jersey.

Schank, R. C. (1975). Theoretical Issues in Natural Language Processing: The Primitive ACTs of Conceptual Dependency. *ACL Anthology*, 34-37.

Smith, N. A. (2002). From Words to Corpora: Recognizing Translation. In *the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 95-102, Philadelphia, PA, USA.

Sopena, J. M., LLoberas, A., & Moliner, J. (1998). A Connectionist Approach to Prepositional Phrase Attachment for Real World Text. In *Proceedings of the Association for Computational Linguistics (ACL)*, pp. 1233-1237, Montreal, Quebec, Canada.

Sowa, J. F. (1983). *Conceptual Structures: Information Processing in Mind and Machine.* Addison-Wesley: MA.

Sowa, J. F. (1992). Conceptual Graphs as a Universal Knowledge Representation. *Computers and Mathematics with Applications*, 23(2-5), 75-93.

Sowa, J. F. (2001). *Conceptual Graphs.* (A working document toward ISO standard: ISO/JTC1/SC 32/WG2). Available at http://www.jfsowa.com/cg/cgstand.htm.

Stede, M.. (1996). *Lexical semantics and knowledge representation in multilingual sentence generation.* Ph.D. thesis, Department of Computer Science, University of Toronto.

Steyvers, M., & Griffiths, T. (2005). Probabilistic topic models. In T. Landauer, D.S. McNamara, S. Dennis, & W. Kintsch (Eds.), *LSA: A Road to Meaning*, Mahwah, NJ: Erlbaum.

Suarez, A., & Palomar, M. (2002). A maximum entropy-based word sense disambiguation system. In Chen, H.-H., & Lin, C.-Y. (Eds.), *Proceedings of the 19th International Conference on Computational Linguistics (COLING),* pp. 960-966.

Swan, M. (1996). *Practical English Usage.* Oxford University Press.

Takahash, T., Iwakura, T., Iida, R., Fujita, A., & Inui, K. (2001). KURA: A Transfer-Based Lexico-Structural Paraphrasing Engine. In *Proceedings of the 6th Natural Language Processing Pacific Rim Symposium (NLPRS), Workshop on Automatic Paraphrasing: Theories and Applications*, Tokyo, Japan.

Towell, G., & Voorhees, E. M. (1998). Disambiguating highly ambiguous words. *Computational Linguistics,* 24(1), 125-145.

University of California, Berkeley. (2000). FrameNet, http://framenet.icsi.berkeley.edu/.

UNL Center of UNDL Foundation. (2005). *Introduction of the UNL System*. Available at http://www.undl.org/unlsys/introduction.html.

USCA Writing Room. (2002). *Paraphrasing*. Available at http://www.usca.edu/writingroom/handouts/paraphrasing.html.

Uzuner, O., Katz, B., & Nahnsen, T. (2005). Using Syntactic Information to Identify Plagiarism. In *Proceeding of the 2nd Workshop on Building Educational Applications using NLP*, pp 37-44, Ann Arbor, MI, USA.

Vinogradovas, M. (2002). The Notions of "Schemata" and "Schemas" in the study of literary fictions. *Respectus Philologicus*, 1(6). Vilnius University, Lithuania. Available at http://www.filologija.lt/102/vinograd.htm.

Voorhees, V. M. (1993). Using WordNet to Disambiguate Word Sense for text Retrieval. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 171-180, Pittsburgh, PA.

Wang, D. (1999). *Knowledge Representation in Health Sciences*. Available at http://www.dbmi.columbia.edu/homepages/wandong/KR/krintro.html.

Wikipedia. (2001). *Wikipedia: free encyclopedia that everyone can edit*, http://en.wikipedia.org/wiki/Main_Page.

Wu, H., & Furugori, T. (1996). A Hybrid Disambiguation Model for Prepositional Phrase Attachment. *Literary and Linguistic Computing,* 11(4), 187-192.

Yang, F., Kim, J., Glass, M., & Evens, M. (2000). Lexical Usage in the Tutoring Schemata of CIRCSIM-Tutor: Analysis of Variable References and Discourse Markers. *Fifth Annual Conference on Human Interaction with Complex Systems (HICS)*, pp. 27-31, Urbana, IL, USA.

Yarowsky, D. (1994). Decision lists for lexical ambiguity resolution: Application to accent restoration in Spanish and French. *In Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 88-95, Las Cruces, NM, USA.

Zhang, Y., & Yamamoto, K. (2002). Paraphrasing of Chinese Utterances. In *Proceeding of the 19th International Conference on Computational Linguistics (COLING),* pp. 1163-1169. Taipei, Taiwan.

Zhang, Y., & Yamamoto, K. (2001). Analysis of Chinese Spoken Language for Automatic Paraphrasing. In *Proceeding of 19th International Conference on Computer Processing of Oriental Language (ICCPOL),* pp. 290-293, Seoul, Korea.

APPENDIX A

MAPPING RULES

This appendix shows the current rules to map a Link triplet to a SSG triplet based on the Link connector type. There are two mapping sets: one is for single-type mapping and another is for multiple-type mapping. A single-type mapping means that one Link triplet will be converted to one SSG triplet. A multiple-type mapping is for several Link triplets that are combined together to create a single SSG triplet.

Single-Type Mapping

Conceptual Relation	**Link Connector Type(s)** **Based on [L, R, type]**
N/A	X (punctuation), W (main clause with left-wall).
Article R -> (Article) -> L	D (determiners to nouns), DD ('the' with proper-nouns)
Agent R -> (Agent) -> L	S (subject-nouns to finite-verbs)
Patient L -> (Agent) -> R	O (transitive-verbs to direct/indirect objects)
Attribute R -> (Attribute) -> L	A (pre-noun adjectives to nouns), AF (adjectives to verbs), AN (noun-modifiers to nouns), E (verb-modifying adverbs to verbs), EA (adverbs to adjectives), EC (adverbs to comparative adjectives), EE (adverbs to other adverbs), EF ('enough' to adjectives and adverbs), EI (adverbs to 'after' and 'before'), EL (some words to 'else' – someone else, what else, etc), EZ (adverbs to 'as' – almost as), Ma (nouns to post-nominal modifiers without comma), Pa ('be' to adjectives)
Attribute L -> (Attribute) -> R	EB (adverbs to 'be' before object, adjective, or prepositional phrase).

Conceptual Relation	Link Connector Type(s) Based on [L, R, type]
Infinitive_Attr R -> (Infinitiive_Attr) -> L	I (verbs with infinitives).

Multi-Type Mapping

Conceptual Relation	Link Connector Type(s) Based on [L1, R1, type] + [L2, R2, type]
Agent L1 -> (Agent) -> R2	MVp + Jp (by)
Time L1 -> (Time) -> R2	MVp + Jp (in)
Patient L1 -> (Patient) -> R2	MVp + Jp
Patient R2 -> (Patient) -> L1	S + Pv [passive voice]
Attribute L1 -> (Attribute) -> R2	Mp + Js/Jp

This list is not exhaustive, but covers those most commonly generated by the Link Grammar. If there are any special cases for which there is no rule defined, the Link Grammar Connector is then used as a relation for a single-type mapping.

APPENDIX B

PRIMITIVE RELATIONS

This Appendix provides a list of primitive relations and their definition.

"*Agent*" in [C1] → (Agent) → [C2] indicates that [C2] is an actor or agent who does an action [C1], or who experiences an action [C1]. Most syntactic-subject of a sentence is an agent, except those in passive-vice sentences. [C2] can be animates or inanimate, including abstract agents, and should be able to does an action (either direct or indirect force).

"*Attribute*" in [C1] → (Attribute) → [C2] indicates that [C2] is an attribute of a situation [C1]. This also includes a situation [C1] is completed, [C2] became a property of an object participated in this situation [C1].

"*Content*" in [C1] → (Content) → [C2] indicates that [C2] is used to indicate the context or content of [C1].

"*Has-Part*" in [C1] → (Has-Part) → [C2] indicates that [C2] has a part [C1].

"*Instrument*" in [C1] → (Instrument) → [C2] indicates that [C2] is a tool used in a situation [C1]. The instrument [C2] includes tangible and intangible objects, and abstract objects.

"*Intention*" in [C1] → (Intention) → [C2] indicates that a situation [C1] was intended to cause [C2] or to make [C2] happen.

"*Is-A*" in [C1] → (Is-A) → [C2] indicates that [C2] is a kind of [C1].

"*Is-Part*" in [C1] → (Is-Part) → [C2] indicates that [C2] is a part of [C1].

"*Location_Above_Surface*" in [C1] → (Location_Above_Surface) → [C2] indicates that [C1] is located *above* a surface of [C2], but does not touch it. The height of [C1] from [C2] may be varied and that can be described by (x, y, z) co-ordinates; where z is the height measured from the surface. Hence, when two different situations have *Location_Above_Surface*, they distinguish one from another by additional information besides a preposition.

"*Location_Destination*" in [C1] → (Location_Destination) → [C2] indicates that [C1] was located that at one place and now its location is a point [C2] (*ending point*).

"*Location_Direction*" in [C1] → (Location_Direction) → [C2] indicates that [C1] is located in a *direction* relative to [C2]. This direction will be replaced by a direction predicates: *North, South, East, West, and combinations*. If two situations mention the same direction, then the distance *d* is used to differentiate how far apart of these two locations.

"*Location_In*" in [C1] → (Location_In) → [C2] indicates that [C1] is located *inside* [C2], which could be an opened/closed container or abstract container. An exact location of [C1] can be described in (x, y, z) co-ordinates relative to the interior of [C2]. And again, two different situations are distinguished by additional information besides a preposition.

"*Location_On_Surface*" in [C1] → (Location_On_Surface) → [C2] indicates that [C1] is located *above* and *touches* a surface of [C2]. If two different situations have the same *Location_On_Surface* predicate, then they both have something else other than a preposition to distinguish them. For example, (x, y) co-ordinates give an exact on-surface location of [C1] relative to [C2].

"*Location_Point*" in [C1] → (Location_Point) → [C2] indicates the position of [C2] in respect to [C1]. *Location_Point* is a generalized predicates. The location [C2] includes tangible and intangible (abstract) objects in all dimensions (D0-dot or point, D1-line or path, D2-surface, and D3-sphere), any landmark location (*e.g.*, school, city), event location (*e.g.,* meeting, festival), or abstract location (*e.g.*, border line, *the line* indicating rules or regulation).

"*Location_Source*" in [C1] → (Location_Source) → [C2] indicates that [C1] was first located at point [C2] (*starting point*) and now its location is another place.

"*Location_Thru*" in [C1] → (Location_Thru) → [C2] indicates that [C1] was located that at point *a* and will be at point *b*, but while changing the location from *a* to *b*, it does pass through a point [C2] (*intermediate point*).

"*Location_Under*" in [C1] → (Location_Under) → [C2] indicates that [C1] is located *below* [C2], but may or may not touch the bottom of [C2]. The touching bottom surface can be determined using the *z* values in (x, y, z) coordinate. When *z* is zero, then [C1] is touching [C2]; otherwise, *z* is a distance that [C1] below [C2].

"*Manner*" in [C1] → (Manner) → [C2] indicates that [C2] is a manner of a situation [C1]. Hence, it could be used as "*Attribute*" of the situation.

"*Patient*" in [C1] → (Patient) → [C2] indicates that [C2] is an object or patient whom receives an action [C1]. Most syntactic-object of a sentence is a patient, except those in passive-voice. [C2] can be animates or inanimate, including abstract agents, and can be either direct or indirect patient.

"*Quantity*" in [C1] → (Quantity) → [C2] indicates that [C2] is used to indicate the quantity of [C1].

"*Time_Destination*" in [C1] → (Time_Destination) → [C2] indicates that [C1] was started some time in the past and now it ends at time [C2] (*ending time*).

"*Time_Duration*" in [C1] → (Time_Duration) → [C2] indicates that situation [C1] was occurred at time *a* and ended at time *b*, with a total of [C2] (*duration for completion*).

"*Time_Interval*" in [C1] → (Time_Thru) → [C2] indicates that situation [C1] occurs at the interval or frequency of [C2].

"*Time_Point*" in [C1] → (Time_Point) → [C2] indicates that the event [C1] occur at time [C2]. *Time_Point* is a generalized predicates. The time [C2] includes real time and abstract time as well as event describing time (*e.g.* in the meeting).

"*Time_Source*" in [C1] → (Time_Source) → [C2] indicates that [C1] was first started at time [C2] (*starting time*) and now its ending is at another point in time.

"*Time_Thru*" in [C1] → (Time_Thru) → [C2] indicates that [C1] was occurred at time *a* and will continue until time *b*, but while the action is in progress during time *a* to *b*, it does pass through a different point in time [C2] (*intermediate point in time*).

"*Verb_Prep*" in [C1] → (Verb_Prep) → [C2] indicates that [C1] is connected to [C2] via a preposition.

APPENDIX C

PREPOSITION USAGE-CASES DEFINITION

To demonstration how each preposition's usage-case is defined for this work, the table below contains a subset of usage-cases definition of eight prepositions: of, to, for, on, at, by, from, and over. Usage-cases of preposition "with" and "in" are shown in Table 7.

Prep	Category	Usage-Case	Head	Complement	HC Relation	SSG Mapping
of	Participant	OfPartGen		person		Attribute
of	Participant	OfPart	person	person		Attribute
of	Location	OfLocDir_P	direction	place		Location_Direction
of	Location	OfLocDir_L	direction	location		Location_Direction
of	Time	OfTimeAt	time_period	month		Time
of	Instrument	OfInstr		disease		Instrument
of	Instrument	OfInstrMatr_T	physical_object	material		Instrument
of	Identification	OfIdentPossObj	physical_object	person		Attribute
of	Identification	OfIdentIsPart	physical_object	physical_object	Is-Part_Comp	Attribute
of	Quality	OfQual_(Q)		quantity		Quality
of	Quality	OfQual_(G)		group		Quality
to	Participant	ToPartRcpt		person		Patient
to	Participant	ToPartPoss		physical_object		Agent
to	Location	ToLocTo_L		location		Location_Destination
to	Location	ToLocTo_A		area		Location_Destination
to	Time	ToTimeTo		time_period		Time_Destination
to	Time	ToTime_T		time		Time_Destination
to	Intention	ToIntenGen	person	human_action		Intention
to	Identification	ToIdent	physical_object	physical_object		Attribute
to	Identification	ToIdentPhys	physical_object	property		Attribute
to	Intention	ToInten_(A)		act		Manner

Prep	Category	Usage-Case	Head	Complement	HC Relation	SSG Mapping
to	Intention	ToInten_(P)		psychological_feature		Manner
from	Participant	FromPartGen	from	person		Agent_2nd
from	Location	FromLocSrc_L	from	location		Location_Source
from	Location	FromLocSrc_A	from	area		Location_Source
from	Intention	FromInten_Verb		verb		Manner
from	Intention	FromInten_Adj	adjective			Manner
from	Identification	FromIdentPers_PP	person_property	person		Attribute
from	Identification	FromIdentPers_PQ	person_quality	person		Attribute
from	Location	FromLoc_(Sky)		sky		Location_Source
from	Location	FromLoc_(Event)		event		Location_Source
from	Quality	FromQual_(Sub)		substance		Quality
from	Quality	FromQual_(Q)		quantity		Quality
for	Participant	ForPartGen		person		Patient
for	Location	ForLocTo_L		location		Location_Destination
for	Location	ForLocTo_A		area		Location_Destination
for	Time	ForTimeAt		time_peirod		Time
for	Time	ForTimeDur		duration		Time_Duration
for	Identification	ForIdentPhys_P		place		Attribute
for	Identification	ForIdentPhys_Prop		property		Attribute
for	Quality	ForQual		amount		Attribute
for	Location	ForLoc_(Exp)		expanse		Location_Destination
for	Location	ForLoc_(Sky)		sky		Location_Destination
for	Intention	ForInten_(Act)		act		Manner
by	Participant	ByPartAgt	physical_object	person		Attribute
by	Location	ByLocThru_L		location		Location_Thru
by	Location	ByLocThru_A		area		Location_Thru
by	Instrument	ByInst		instrumentality		Instrument
by	Instrument	ByInst_M		medium		Instrument
by	Identification	ByIdentPhys	physical_object	person	author	Attribute
by	Quality	ByQual	number	number		Quality
by	Participant	ByPart_(T)		thing		Agent

Prep	Category	Usage-Case	Head	Complement	HC Relation	SSG Mapping
by	Participant	ByPart_(CA)		causal_agent		Agent
at	Participant	AtPartRcpt		person		Patient
at	Location	AtLocAt_L		location		Location
at	Location	AtLocAt_A		area		Location
at	Time	AtTimeAt_T		time		Time
at	Time	AtTimeAt_S		season		Time
at	Intention	AtInten_Verb		verb		Agent
at	Instrument	AtInstr_I		invitation		Instrument
at	Identification	AtIdentPers_Prop	person	person_property		Attribute
at	Identification	AtIdentPers_S		subject		Attribute
at	Location	AtLoc_(Sky)		sky		Location
at	Location	AtLoc_(Event)		event		Location
over	Participant	OverPartGen_Per		person		Agent
over	Participant	OverPartGen_C		country		Agent
over	Location	OverLocAt_L		location		Location_Destination
over	Location	OverLocAt_A		area		Location_Destination
over	Time	OverTimeAt_T		time		Time_At
over	Time	OverTimeAt_H		holiday		Time_At
over	Time	OverTimeAt_S		season		Time_At
over	Time	OverTimeDur_M		meal		Time_Dur
over	Instrument	OverInstr		instrumentality		Instrument
over	Instrument	OverInstr_M		medium		Instrument
over	Quality	OverQual_(Sub)		substance		Quality_over
over	Quality	OverQual_(G)		group		Quality_over
on	Participant	OnPartGen		person		Patient
on	Location	OnLocAt_L		location		Location
on	Location	OnLocAt_A		area		Location
on	Location	OnLocAt_P		place		Location
on	Location	OnLocDir		direction		Location_Dir
on	Time	OnTimeAt_D		date		Time
on	Time	OnTimeAt_TP		time_period		Time
on	Instrument	OnInstr		instrumentality		Instrument

Prep	Category	Usage-Case	Head	Complement	HC Relation	SSG Mapping
on	Participant	OnPart_(CA)		causal_agent		Patient
on	Quality	OnQual_(Sub)		substance		Quality
on	Quality	OnQual_(G)		group		Quality

APPENDIX D

PARAPHRASE RECOGNITION RULES

This shows various paraphrase recognition rules, similar to the one shown in Figure 32. It is a not a complete and exhaustive list. That is, there are some rules missing. However, with the current implementation, these paraphrase rules are sufficient.

```
ParaRuleDef (
    ParaRuleName: Patient-Inst
    ParaType:  Instrument
    LeftLink:  ([VERB.v] -> (Patient) -> [INSTRUMENTALITY.n])
    RightLink: ([VERB.v] -> (Instrument) ->
[INSTRUMENTALITY.n])
)

ParaRuleDef (
    ParaRuleName: Attr-Manner
    ParaType: Manner
    LeftLink: ([VERB.v] -> (Attribute) -> [ADVERB.adv])
    RightLink: ([VERB.v] -> (Manner) -> [NOUN.n])
    Rel: ([ADVERB.adv], [NOUN.n], [same lemma])
)

ParaRuleDef (
    ParaRuleName: USE-TO-Do-Manner-Inst
    ParaType: Instrument
    LeftLink:  ([VERB.v] -> (Agent) -> [AGENT.n];
                [VERB.v] -> (Instrument) -> [INSTRUMENTALITY.n])
    RightLink: ([VERB.v] -> (Agent) -> [AGENT.n];
                [use.v] -> (Manner) -> [VERB.v];
                [use.v] -> (Patient) -> [INSTRUMENTALITY.n])
)

ParaRuleDef (
    ParaRuleName: USE-TO-Do-Attribute-Inst
    ParaType: Instrument
    LeftLink: ([VERB.v] -> (Agent) -> [AGENT.n];
               [VERB.v] -> (Instrument) -> [INSTRUMENTALITY.n])
    RightLink: ([VERB.v] -> (Agent) -> [AGENT.n];
                [use.v] -> (Attribute) -> [VERB.v];
                [use.v] -> (Patient) -> [INSTRUMENTALITY.n])
)
```

```
ParaRuleDef (
    ParaRuleName: Has-Part,Is-Part
    ParaType: Has-Part
    LeftLink: ([VERB.v] -> (Patient) -> [NOUN_1.n];
               [NOUN_1.n] -> (Has-Part) -> [NOUN_2.n])
    RightLink: ([VERB.v] -> (Patient) -> [NOUN_2.n];
               [NOUN_2.n] -> (Is-Part) -> [NOUN_1.n])
    Rel:([NOUN_1.n], [NOUN_2.n], [hypernym])
)

ParaRuleDef (
    ParaRuleName: Has-Part-Patient
    ParaType: Has-Part
    LeftLink: ([NOUN_1.n] -> (Has-Part) -> [NOUN_2.n])
    RightLink: ([has.v] -> (Agent) -> [NOUN_1.n];
               [has.v] -> (Patient) -> [NOUN_2.n])
    Rel:([NOUN_1.n], [NOUN_2.n], [hypernym])
)

ParaRuleDef (
    ParaRuleName: Has-Part-Attribute
    ParaType: Has-Part
    LeftLink: ([NOUN_1.n] -> (Has-Part) -> [NOUN_2.n])
    RightLink: ([NOUN_1.n] -> (Attribute) -> [NOUN_2.n])
    Rel:([NOUN_1.n], [NOUN_2.n], [hypernym])
)

ParaRuleDef (
    ParaRuleName: Has-Attribute
    ParaType: Has-Attribute
    LeftLink: ([NOUN_1.n] -> (Attribute) -> [NOUN_2.n])
    RightLink: ([has.v] -> (Agent) -> [NOUN_1.n];
               [has.v] -> (Patient) -> [NOUN_2.n])
    Rel:([NOUN_1.n], [NOUN_2.n], [hypernym])
)

ParaRuleDef (
    ParaRuleName: Attribute-Manner-Verb
    ParaType: Attribute-Manner
    LeftLink: ([VERB_1.n] -> (Attribute) -> [VERB_2.n])
    RightLink: ([VERB_1.n] -> (Manner) -> [VERB_2.n])
)

ParaRuleDef (
    ParaRuleName: MoreThan-Over-Patient
    ParaType: Patient
    LeftLink: ([VERB.v] -> (Patient) -> [NOUN.n])
    RightLink: ([VERB.v] -> (MVt) -> [than];
               [VERB.v] -> (MVm) -> [more];
               [VERB.v] -> (Patient) -> [NOUN.n])
)
```

```
ParaRuleDef (
    ParaRuleName: Attribute-Adj
    ParaType: Attribute
    LeftLink: ([NOUN_1.n] -> (Attribute) -> [NOUN_2.n])
    RightLink: ([NOUN_1.n] -> (Attribute) -> [ADJECTIVE.adj])
)
```

VITA

CHUTIMA BOONTHUM

School : Department of Computer Science
Old Dominion University
Norfolk, VA-23508

Education Background

Ph.D., 2007, Computer Science, Old Dominion University, Norfolk, VA
M.S., 2000, Applied Computer Science, Illinois State University, Normal, IL
B.S., 1997, Computer Science, Srinakharinwirot University, Bangkok, Thailand

Work Experience

Assistant Professor (September 2006 - present), Department of Computer Science, Hampton University

Research Assistant (Jan 2001 - present), Department of Computer Science, Old Dominion University

Selected Publications

McNamara, D. S., O'Reilly, T., Rowe, M., Boonthum, C., & Levinstein, I. B. (2007). iSTART: A Web-Based Tutor that Teaches Self-Explanation and Metacognitive Reading Strategies. In D. S. McNamara (Ed.), *Reading Comprehension Strategies: Theories, Interventions, and Technologies.*

Boonthum, C., Levinstein, I.B, & McNamara, D. S. (2006) Evaluating Self-Explanations in iSTART: Word Matching, Latent Semantic Analysis, and Topic Models. In A. Kao & S. Poteet (Eds.), *Text Mining and Natural Language Processing*, Springer. 91-106.

Boonthum, C., Toida, S., & Levinstein, I. B. (2006) Preposition Senses: Generalized Disambiguation Model. *Proceedings of the seventh International Conference on Computational Linguistics and Intelligent Text Processing (CICLing-2006)*, Lecture Notes in Computer Science, Springer Verlag GmbH. 196-207.

Boonthum, C., Toida, S., & Levinstein, I. B. (2005). Sense Disambiguation for Preposition 'with'. *Proceedings of the Second ACL-SIGSEM Workshop on "The Linguistic Dimensions of Prepositions and their Use in Computational Linguistic Formalisms and Applications"*, University of Essex – Colchester, United Kingdom. 153-162.

Boonthum, C. (2004). iSTART: Paraphrase Recognition. *Proceedings of the Student Research Workshop: ACL 2004. 42nd Annual Meeting of the Association of Computational Linguistics*, Barcelona, Spain. 31-36.