

Semi-Automatic Recognition of Noun Modifier Relationships

Ken BARKER and Stan SZPAKOWICZ
School of Information Technology and Engineering
University of Ottawa
Ottawa, Canada K1N 6N5
{kbarker, szpak}@site.uottawa.ca

Abstract

Semantic relationships among words and phrases are often marked by explicit syntactic or lexical clues that help recognize such relationships in texts. Within complex nominals, however, few overt clues are available. Systems that analyze such nominals must compensate for the lack of surface clues with other information. One way is to load the system with lexical semantics for nouns or adjectives. This merely shifts the problem elsewhere: how do we define the lexical semantics and build large semantic lexicons? Another way is to find constructions *similar* to a given complex nominal, for which the relationships are already known. This is the way we chose, but it too has drawbacks. Similarity is not easily assessed, similar analyzed constructions may not exist, and if they do exist, their analysis may not be appropriate for the current nominal.

We present a semi-automatic system that identifies semantic relationships in noun phrases without using precoded noun or adjective semantics. Instead, partial matching on previously analyzed noun phrases leads to a tentative interpretation of a new input. Processing can start without prior analyses, but the early stage requires user interaction. As more noun phrases are analyzed, the system learns to find better interpretations and reduces its reliance on the user. In experiments on English technical texts the system correctly identified 60-70% of relationships automatically.

1 Introduction

Any system that extracts knowledge from text cannot ignore complex noun phrases. In technical domains especially, noun phrases carry much of the information. Part of that information is contained in words; cataloguing the semantics of sin-

gle words for computational purposes is a difficult task that has received much attention. But part of the information in noun phrases is contained in the relationships between components.

We have built a system for noun modifier relationship (NMR) analysis that assigns semantic relationships in complex noun phrases. Syntactic analysis finds noun phrases in a sentence and provides a flat list of premodifiers and postmodifying prepositional phrases and appositives. The NMR analyzer first brackets the flat list of premodifiers into modifier-head pairs. Next, it assigns NMRs to each pair. NMRs are also assigned to the relationships between the noun phrase and each post-modifying phrase.

2 Background

2.1 Noun Compounds

A head noun along with a noun premodifier is often called a noun compound. Syntactically a noun compound acts as a noun: a modifier or a head may again be a compound. The NMR analyzer deals with the semantics of a particular kind of compound, namely those that are *transparent* and *endocentric*.

The meaning of a *transparent* compound can be derived from the meaning of its elements. For example, *laser printer* is transparent (a *printer* that uses a *laser*). *Guinea pig* is *opaque*: there is no obvious direct relationship to *guinea* or to *pig*.

An *endocentric* compound is a hyponym of its head. *Desktop computer* is endocentric because it is a kind of *computer*. *Bird brain* is *exocentric* because it does not refer to a kind of *brain*, but rather to a kind of person (whose brain resembles that of a bird).

Since the NMR analyzer is intended for technical texts, the restriction to transparent endocentric compounds should not limit the utility of the system. Our experiments have found no opaque or exocentric compounds in the test texts.

2.2 Semantic Relations in Noun Phrases

Most of the research on relationships between nouns and modifiers deals with noun compounds, but these relationships also hold between nouns and adjective premodifiers or postmodifying prepositional phrases. Lists of semantic labels have been proposed, based on the theory that a compound expresses one of a small number of covert semantic relations.

Levi (1978) argues that semantics and word formation make noun-noun compounds a heterogeneous class. She removes opaque compounds and adds nominal non-predicating adjectives. For this class Levi offers nine semantic labels. According to her theory, these labels represent underlying predicates deleted during compound formation. George (1987) disputes the claim that Levi's non-predicating adjectives never appear in predicative position.

Warren (1978) describes a multi-level system of semantic labels for noun-noun relationships. Warren (1984) extends the earlier work to cover adjective premodifiers as well as nouns. The similarity of the two lists suggests that many adjectives and premodifying nouns can be handled by the same set of semantic relations.

2.3 Recognizing Semantic Relations

Programs that uncover the relationships in modifier-noun compounds often base their analysis on the semantics of the individual words (or a composition thereof). Such systems assume the existence of some semantic lexicon.

Leonard's system (1984) assigns semantic labels to noun-noun compounds based on a dictionary that includes taxonomic and meronymic (part-whole) information, information about the syntactic behaviour of nouns and about the relationships between nouns and verbs. Finin (1986) produces multiple semantic interpretations of modifier-noun compounds. The interpretations are based on pre-coded semantic class information and domain-dependent frames describing the roles that can be associated with certain nouns. Ter Stal's system (1996) identifies concepts in text and unifies them with structures extracted from a hand-coded lexicon containing syntactic information, logical form templates and taxonomic information.

In an attempt to avoid the hand-coding required in other systems, Vanderwende (1993)

automatically extracts semantic features of nouns from online dictionaries. Combinations of features imply particular semantic interpretations of the relationship between two nouns in a compound.

3 Noun Modifier Relationship Labels

Table 1 lists the NMRs used by our analyzer. The list is based on similar lists found in literature on the semantics of noun compounds. It may evolve as experimental evidence suggests changes.

Agent (agt)	Material (matr)
Beneficiary (benf)	Object (obj)
Cause (caus)	Possessor (poss)
Container (ctn)	Product (prod)
Content (cont)	Property (prop)
Destination (dest)	Purpose (purp)
Equative (equa)	Result (resu)
Instrument (inst)	Source (src)
Located (led)	Time (time)
Location (loc)	Topic (top)

Table 1: The noun modifier relationships

For each NMR, we give a paraphrase and example modifier-noun compounds. Following the tradition in the study of noun compound semantics, the paraphrases act as definitions and can be used to check the acceptability of different interpretations of a compound. The paraphrases serve as definitions in this section and to help with interpretation during user interactions (as illustrated in section 6). In the analyzer, awkward paraphrases with adjectives could be improved by replacing adjectives with their WordNet *pertainyms* (Miller, 1990), giving, for example, "charity benefits from charitable donation" instead of "charitable benefits from charitable donation".

Agent: compound *is performed by* modifier
student protest, band concert, military assault

Beneficiary: modifier *benefits from* compound
student price, charitable donation

Cause: modifier *causes* compound
exam anxiety, overdue fine

Container: modifier *contains* compound
printer tray, flood water, film music, story idea

Content: modifier *is contained in* compound
paper tray, eviction notice, oil pan

Destination: modifier *is destination of* compound
game bus, exit route, entrance stairs

Equative: modifier *is also* head
composer arranger, player coach

Instrument: modifier *is used in* compound
electron microscope, diesel engine, laser printer

Located: modifier *is located at* compound
building site, home town, solar system

Location: modifier *is the location of* compound
lab printer, internal combustion, desert storm

Material: compound *is made of* modifier
carbon deposit, gingerbread man, water vapour

Object: modifier *is acted on by* compound
engine repair, horse doctor

Possessor: modifier *has* compound
national debt, student loan, company car

Product: modifier *is a product of* compound
automobile factory, light bulb, colour printer

Property: compound *is* modifier
blue car, big house, fast computer

Purpose: compound *is meant for* modifier
concert hall, soup pot, grinding abrasive

Result: modifier *is a result of* compound
storm cloud, cold virus, death penalty

Source: modifier *is the source of* compound
foreign capital, chest pain, north wind

Time: modifier *is the time of* compound
winter semester, late supper, morning class

Topic: compound *is concerned with* modifier
computer expert, safety standard, horror novel

4 Noun Modifier Bracketing

Before assigning NMRs, the system must bracket the head noun and the premodifier sequence into modifier-head pairs. Example (2) shows the bracketing for noun phrase (1).

- (1) *dynamic high impedance microphone*
 (2) *(dynamic ((high impedance) microphone))*

The bracketing problem for noun-noun-noun compounds has been investigated by Liberman & Sproat (1992), Pustejovsky *et al.* (1993), Resnik (1993) and Lauer (1995) among others. Since the NMR analyzer must handle premodifier sequences of any length with both nouns and adjectives, it requires more general techniques. Our semi-automatic bracketer (Barker, 1998) allows for any number of adjective or noun premodifiers.

After bracketing, each non-atomic element of a bracketed pair is considered a subphrase of the original phrase. The subphrases for the bracketing in (2) appear in (3), (4) and (5).

- (3) *high impedance*
 (4) *high_impedance microphone*
 (5) *dynamic high_impedance_microphone*

Each subphrase consists of a modifier (possibly compound, as in (4)) and a head (possibly compound, as in (5)). The NMR analyzer assigns an NMR to the modifier-head pair that makes up each subphrase.

Once an NMR has been assigned, the system must store the assignment to help automate future processing. Instead of memorizing complete noun phrases (or even complete subphrases) and analyses, the system reduces compound modifiers and compound heads to their own local heads and stores these *reduced* pairs with their assigned NMR. This allows it to analyze different noun phrases that have only reduced pairs in common with previous phrases. For example, (6) and (7) have the reduced pair (8) in common. If (6) has already been analyzed, its analysis can be used to assist in the analysis of (7)—see section 5.1.

- (6) *(dynamic ((high impedance) microphone))*
 (7) *(dynamic (cardioid (vocal microphone)))*
 (8) *(dynamic microphone)*

5 Assigning NMRs

Three kinds of construction require NMR assignments: the modifier-head pairs from the bracketed premodifier sequence; postmodifying prepositional phrases; appositives.

These three kinds of input can be generalized to a single form—a triple consisting of modifier, head and marker (M, H, Mk). For premodifiers, Mk is the symbol nil, since no lexical item links the premodifier to the head. For postmodifying prepositional phrases Mk is the preposition. For appositives, Mk is the symbol appos. The (M, H, Mk) triples for examples (9), (10) and (11) appear in Table 2.

- (9) *monitor cable plug*
 (10) *large piece of chocolate cake*
 (11) *my brother, a friend to all young people*

To assign an NMR to a triple (M, H, Mk), the system looks for previous triples whose distance to the current triple is minimal. The NMRs assigned to previous similar triples comprise lists of candidate NMRs. The analyzer then finds what it considers the best NMR from these lists of candi-

<i>Modifier</i>	<i>Head</i>	<i>Marker</i>
monitor	cable	nil
monitor_cable	plug	nil
chocolate	cake	nil
large	piece	nil
chocolate_cake	large_piece	of
young	people	nil
young_people	friend	to
friend	brother	appos

Table 2: (M, H, Mk) triples for (9), (10) and (11)

dates to present to the user for approval. Appositives are automatically assigned Equative.

5.1 Distance Between Triples

The distance between two triples is a measure of the degree to which their modifiers, heads and markers match. Table 3 gives the eight different values for distance used by NMR analysis.

The analyzer looks for previous triples at the lower distances before attempting to find triples at higher distances. For example, it will try to find identical triples before trying to find triples whose markers do not match.

Several things about the distance measures require explanation. First, a preposition is more similar to a nil marker than to a different preposition. Unlike a different preposition, the nil marker is not known to be different from the marker in an overtly marked pair.

Next, no evidence suggests that triples with matching M are more similar or less similar than triples with matching H (distances 3 and 6).

Triples with matching prepositional marker (distance 4) are considered more similar than triples with matching M or H only. A preposition is an overt indicator of the relationship between M and H (see Quirk, 1985: chapter 9) so a correlation is more likely between the preposition and the

NMR than between a given M or H and the NMR.

If the current triple has a prepositional marker not seen in any previous triple (distance 5), the system finds candidate NMRs in its NMR marker dictionary. This dictionary was constructed from a list of about 50 common atomic and phrasal prepositions. The various meanings of each preposition were mapped to NMRs by hand. Since the list of prepositions is small, dictionary construction was not a difficult knowledge engineering task (requiring just twenty hours of work of a secondary school student).

5.2 The Best NMRs

The lists of candidate NMRs consist of all those NMRs previously assigned to (M, H, Mk) triples at a minimum distance from the triple under analysis. If the minimum distance was 3 or 6, there may be two candidate lists: L_M contains the NMRs previously assigned to triples with matching M, L_H with matching H. The analyzer attempts to choose a set R of candidates to suggest to the user as the best NMRs for the current triple.

If there is one list L of candidate NMRs, R contains the NMR (or NMRs) that occur most frequently in L . For two lists L_M and L_H , R could be found in several ways. We could take R to contain the most frequent NMRs in $L_M \cup L_H$. This absolute frequency approach has a bias towards NMRs in the larger of the two lists.

Alternatively, the system could prefer NMRs with the highest *relative* frequency in their lists. If there is less variety in the NMRs in L_M than in L_H , M might be a more consistent indicator of NMR than H. Consider example (12).

(12) *front line*

Compounds with the modifier *front* may always have been assigned Location. Compounds with

<i>dist</i>	<i>current triple</i>	<i>previous triple</i>	<i>example</i>	
0	(M, H, Mk)	(M, H, Mk)	<i>wall beside a garden</i>	<i>wall beside a garden</i>
1	(M, H, <prep>)	(M, H, nil)	<i>wall beside a garden</i>	<i>garden wall</i>
2	(M, H, Mk)	(M, H, _)	<i>wall beside a garden</i>	<i>wall around a garden</i>
3	(M, H, Mk)	(M, _, Mk) or (_, H, Mk)	<i>pile of garbage</i>	<i>pile of sweaters</i>
4	(M, H, <prep>)	(_, _, <prep>)	<i>pile of garbage</i>	<i>house of bricks</i>
5	(M, H, <prep>)	(_, _, _)	<i>ice in the cup</i>	nmmr(in, [ctn,inst,loc,src,time])
6	(M, H, Mk)	(M, _, _) or (_, H, _)	<i>wall beside a garden</i>	<i>garden fence</i>
7	(M, H, Mk)	(_, _, _)	<i>wall beside a garden</i>	<i>pile of garbage</i>

Table 3: Measures of distance between triples

the head *line* may have been assigned many different NMRs. If *line* has been seen as a head more often than *front* as a modifier, one of the NMRs assigned to *line* may have the highest absolute frequency in $\mathbf{L}_M \cup \mathbf{L}_H$. But if Location has the highest relative frequency, this method correctly assigns Location to (12). There is a potential bias, however, for smaller lists (a single NMR in a list always has the highest relative frequency).

To avoid these biases, we could combine absolute and relative frequencies. Each NMR i is assigned a score s_i calculated as:

$$s_i = \frac{\text{freq}(i \in \mathbf{L}_M)^2}{|\mathbf{L}_M|} + \frac{\text{freq}(i \in \mathbf{L}_H)^2}{|\mathbf{L}_H|}$$

\mathbf{R} would contain NMR(s) with the highest score. This combined formula was used in the experiment described in section 7.

5.3 Premodifiers as Classifiers

Since NMR analysis deals with endocentric compounds we can recover a taxonomic relationship from triples with a nil marker. Consider example (13) and its reduced pairs in (14):

(13) ((*laser printer*) *stand*)

(14) (*laser printer*)
(*printer stand*)

These pairs produce the following output:

laser_printer_stand isa *stand*
laser_printer isa *printer*

6 User Interaction

The NMR analyzer is intended to start processing from scratch. A session begins with no previous triples to match against the triple at hand. To compensate for the lack of previous analyses, the system relies on the help of a user, who supplies the correct NMR when the system cannot determine it automatically.

In order to supply the correct NMR, or even to determine if the suggested NMR is correct, the user must be familiar with the NMR definitions. To minimize the burden of this requirement, all interactions use the modifier and head of the current phrase in the paraphrases from section 3. Furthermore, if the appropriate NMR is not among those suggested by the system, the user can request the complete list of paraphrases with the current modifier and head.

6.1 An Example

Figure 1 shows the interaction for phrases (15)-(18). The system starts with no previously analyzed phrases. The NMR marker dictionary maps the preposition *of* to twelve NMRs: Agent, Cause, Content, Equative, Located, Material, Object, Possessor, Property, Result, Source, Topic.

(15) *small gasoline engine*

(16) *the repair of diesel engines*

(17) *diesel engine repair shop*

(18) *an auto repair center*

User input is shown bold underlined. At any prompt the user may type ‘list’ to view the complete list of NMR paraphrases for the current modifier and head.

7 Evaluation

We present the results of evaluating the NMR analyzer in the context of a large knowledge acquisition experiment (see Barker *et al.*, 1998). The NMR analyzer is one part of a larger interactive semantic analysis system. The experiment evaluated the semantic analysis of Atkinson (1990). We refer to it as the *small engines* experiment. Other experiments have shown similar results.

We consider three evaluation criteria. First, we evaluate the analyzer’s ability to learn to make better suggestions to the user as more noun phrases are analyzed. Second, we evaluate its coverage by comparing the number of relationships assigned with the total number of such relationships in the text (*i.e.*, the number it *should have* assigned). Third, we assess the burden that semi-automatic analysis places on the user.

7.1 Improvement in System Performance

Since the system starts with no previous noun phrase analyses, the user is responsible for supplying NMRs at the beginning of a session. To measure the rate of learning, we compare the cumulative number of assignments required from the user to the cumulative number of correct assignments suggested by the system.

In the *small engines* experiment, 886 modifier-noun pairs were assigned an NMR. We consider the system’s assignment correct when the correct label is among its suggestions. According to this definition, 608 of the 886 NMRs (69%) were assigned correctly by the system. For most

of these assignments (97.5%) the system offered a single suggestion. It had multiple (on average 3.3) suggestions only 22 times.

<i>Phrase (15): small gasoline engine</i>	
There is a relationship between <i>gasoline</i> and <i>gasoline_engine</i> .	
Please enter a valid NMR label:	<u>inst</u>
Do you accept the NMR Instrument: <i>gasoline</i> is used in <i>gasoline_engine</i>	<u>Y</u>
There is a relationship between <i>small</i> and <i>small_gasoline_engine</i> .	
Please enter a valid NMR label:	<u>prop</u>
Do you accept the NMR Property: <i>small_gasoline_engine</i> is <i>small</i>	<u>Y</u>
<i>Phrase (16): the repair of diesel engines</i>	
There is a relationship between <i>diesel</i> and <i>diesel_engine</i> .	
NMR Analyzer's best suggestions for this input:	
(1) prop: <i>diesel_engine</i> is <i>diesel</i>	
(2) inst: <i>diesel</i> is used in <i>diesel_engine</i>	
Please enter a number between 1 and 2:	<u>2</u>
Do you accept the NMR Instrument: <i>diesel</i> is used in <i>diesel_engine</i>	<u>Y</u>
There is a relationship between <i>diesel_engine</i> and <i>repair</i> .	
NMR Analyzer's best suggestions for this input:	
(1) agt: <i>repair</i> is performed by <i>diesel_engine</i>	
(2) caus: <i>diesel_engine</i> causes <i>repair</i>	
...	
(7) obj: <i>diesel_engine</i> is acted on by <i>repair</i>	
...	
(12) top: <i>repair</i> is concerned with <i>diesel_engine</i>	
Please enter a number between 1 and 12:	<u>7</u>
Do you accept the NMR Object: <i>diesel_engine</i> is acted on by <i>repair</i>	<u>Y</u>
<i>Phrase (17): diesel engine repair shop</i>	
Do you accept the NMR Instrument: <i>diesel</i> is used in <i>diesel_engine</i>	<u>Y</u>
Do you accept the NMR Object: <i>diesel_engine</i> is acted on by <i>diesel_engine_repair</i>	<u>Y</u>
There is a relationship between <i>diesel_engine_repair</i> and <i>diesel_engine_repair_shop</i> .	
Please enter a valid NMR label:	<u>purp</u>
Do you accept the NMR Purpose: <i>diesel_engine_repair_shop</i> is meant for <i>diesel_engine_repair</i>	<u>Y</u>
<i>Phrase (18): an auto repair center</i>	
Do you accept the NMR Object: <i>auto</i> is acted on by <i>auto_repair</i>	<u>Y</u>
Do you accept the NMR Purpose: <i>auto_repair_center</i> is meant for <i>auto_repair</i>	<u>Y</u>

Figure 1: NMR analysis interaction for (15)-(18)

Figure 2 shows the cumulative number of NMR assignments supplied by the user versus those determined correctly by the system. After about 100 assignments, the system was able to make the majority of assignments automatically. The curves in the figure show that the system learns to make better suggestions as more phrases are analyzed.

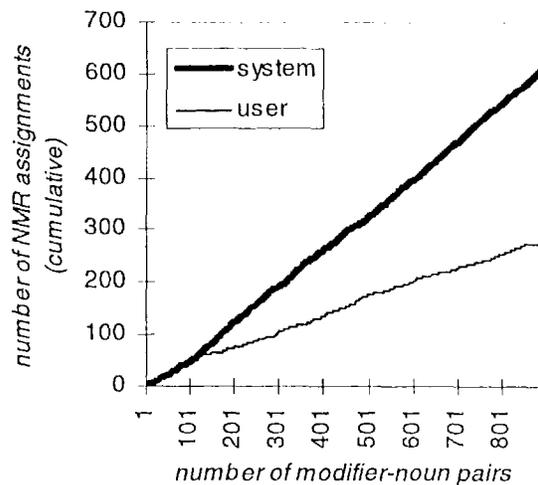


Figure 2: Cumulative NMR assignments

7.2 NMR Coverage

The NMR analyzer depends on a parser to find noun phrases in a text. If parsing is not 100% successful, the analyzer will not see all noun phrases in the input text. It is not feasible to find manually the total number of relationships in a text—even in one of only a few hundred sentences. To measure coverage, we sampled 100 modifier-noun pairs at random from the *small engines* text and found that 87 of them appeared in the analyzer's output. At 95% confidence, we can say that the system extracted between 79.0% and 92.2% of the modifier-noun relationships in the text.

7.3 User Burden

User burden is a fairly subjective criterion. To measure burden, we assigned an “onus” rating to each interaction during the *small engines* experiment. The onus is a number from 0 to 3. 0 means that the correct NMR was obvious, whether suggested by the system or supplied by the user. 1 means that selecting an NMR required a few moments of reflection. A rating of 2 means that the interaction required serious thought, but we were

ultimately able to choose an NMR. 3 means that even after much contemplation, we were unable to agree on an NMR.

The average user onus rating was 0.1 for NMR interactions in the *small engines* experiment. 808 of the 886 NMR assignments received an onus rating of 0; 71 had a rating of 1; 7 received a rating of 2. No interactions were rated onus level 3.

8 Future Work

Although the list of NMRs was inspired by the relationships found commonly in others' lists, it has not undergone a more rigorous validation (such as one described in Barker *et al.*, 1997).

In section 5.2 we discussed different approaches to choosing NMRs from two lists of candidates. We have implemented and compared five different techniques for choosing the best NMRs, but experimental results are inconclusive as to which techniques are better. We should seek a more theoretically sound approach followed by further experimentation.

The NMR analyzer currently allows its stored triples (and associated NMRs) to be saved in a file at the end of a session. Any number of such files can be reloaded at the beginning of subsequent sessions, "seeding" the new sessions. It is necessary to establish the extent to which the triples and assignments from one text or domain are useful in the analysis of noun phrases from another domain.

Acknowledgements

This work is supported by the Natural Sciences and Engineering Research Council of Canada.

References

- Atkinson, Henry F. (1990). *Mechanics of Small Engines*. New York: Gregg Division, McGraw-Hill.
- Barker, Ken (1998). "A Trainable Bracketeer for Noun Modifiers". *The Twelfth Canadian Conference on Artificial Intelligence*.
- Barker, Ken, Terry Copeck, Sylvain Delisle & Stan Szpakowicz (1997). "Systematic Construction of a Versatile Case System." *Journal of Natural Language Engineering* 3(4), December 1997.
- Barker, Ken, Sylvain Delisle & Stan Szpakowicz (1998). "Test-Driving TANKA: Evaluating a Semi-Automatic System of Text Analysis for Knowledge Acquisition." *The Twelfth Canadian Conference on Artificial Intelligence*.
- Finin, Timothy W. (1986). "Constraining the Interpretation of Nominal Compounds in a Limited Context." In *Analyzing Language in Restricted Domains: Sublanguage Description and Processing*, R. Grishman & R. Kittredge, eds., Lawrence Erlbaum, Hillsdale, pp. 163-173.
- George, Steffi (1987). *On "Nominal Non-Predicating" Adjectives in English*. Frankfurt am Main: Peter Lang.
- Lauer, Mark (1995). "Corpus Statistics Meet the Noun Compound: Some Empirical Results." *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*. Cambridge. 47-54.
- Leonard, Rosemary (1984). *The Interpretation of English Noun Sequences on the Computer*. Amsterdam: North-Holland.
- Levi, Judith N. (1978). *The Syntax and Semantics of Complex Nominals*. New York: Academic Press.
- Liberman, Mark & Richard Sproat (1992). "Stress and Structure of Modified Noun Phrases." *Lexical Matters (CSLI Lecture Notes, 24)*. Stanford: Center for the Study of Language and Information.
- Miller, George A., ed. (1990). "WordNet: An On-Line Lexical Database." *International Journal of Lexicography* 3(4).
- Pustejovsky, James, S. Bergler & P. Anick (1993). "Lexical Semantic Techniques for Corpus Analysis." *Computational Linguistics* 19(2). 331-358.
- Quirk, Randolph, Sidney Greenbaum, Geoffrey Leech & Jan Svartvik (1985). *A Comprehensive Grammar of the English Language*. London: Longman.
- Resnik, Philip Stuart (1993). "Selection and Information: A Class-Based Approach to Lexical Relationships." Ph.D. thesis, IRCS Report 93-42, University of Pennsylvania.
- ter Stal, Wilco (1996). "Automated Interpretation of Nominal Compounds in a Technical Domain." Ph.D. thesis, University of Twente, The Netherlands.
- Vanderwende, Lucy (1993). "SENS: The System for Evaluating Noun Sequences." In *Natural Language Processing: The PLNLP Approach*, K. Jensen, G. Heidorn & S. Richardson, eds., Kluwer Academic Publishers, Boston, pp. 161-173.
- Warren, Beatrice (1978). *Semantic Patterns of Noun-Noun Compounds*. Göteborg: Acta Universitatis Gothoburgensis.
- Warren, Beatrice (1984). *Classifying Adjectives*. Göteborg: Acta Universitatis Gothoburgensis.