A TOOL FOR INVESTIGATING THE SYNONYMY RELATION IN A SENSE DISAMBIGUATED THESAURUS

Martin S. Chodorow
IBM T.J. Watson Research Center
Yorktown Heights, New York 10598
and
Department of Psychology, Hunter College of CUNY
New York, New York 10021

Yael Ravin
IBM T.J. Watson Research Center
Yorktown Heights, New York 10598

Howard E. Sachar
IBM Data Systems Division
White Plains, New York 10601

Abstract

This paper describes an exploration of the implicit synonymy relationship expressed by synonym lists in an on-line thesaurus. A series of automatic steps was taken to properly constrain this relationship. The resulting groupings of semantically related word senses are believed to constitute a useful tool for natural language processing and for work in lexicography.

Introduction

The importance of semantic processing of natural language is generally acknowledged (Grishman 1986) and needs no justification. Work on applications such as information retrieval or machine translation has consistently focused on semantic analysis. A wide range of models has been suggested, based on semantic networks, on fuzzy logic, on conceptual dependencies and more. Common to all these models, however, is the researchers' reliance on hand-built semantic databases. These databases tend to be rather limited in scope and often restricted to narrow domains. If the process of constructing them remains manual, broad-coverage semantic analysis by computers will be severely handicapped for quite a long time. It is our goal, therefore, to explore automatic and semi-automatic ways of constructing these semantic databases, through the manipulation of machine-readable semantic sources. In this paper, we concentrate on heuristics for the automatic manipulation of synonyms found in an on-line thesaurus.

First, we should clarify what we mean by "synonyms". The definition of synonymy and the existence of synonyms have long been debated in linguistics. Some believe it is impossible to capture meaning, not even of the most concrete terms in natural language. Consequently, it is impossible to define synonymy or to identify synonymous terms (Quine 1960). Others believe it is possible to give full semantic representations of meaning and therefore to define synonymy formally and to identify true synonyms (Katz and Fodor 1963). According to this view, synonymy is a relationship of sameness of meaning between words, which is defined as the identity of their semantic representations.

We have chosen an operational approach to synonymy: The synonyms of a headword *w* are whatever words are listed in the entry for *w* in an on-line version of *The New Collins Thesaurus* (1984) (CT).[1] According to the authors, "...no synonym is entered unless it is *fully* substitutable for the headword in a sensi-

[1] We have stored CT as a DAM file (Byrd, et al., 1986) with 16,794 keyed records containing a total of 287,136 synonym tokens. It has been supplemented with part-of-speech information from the UDICT computerized lexicon system (Byrd, 1986).

ble English sentence" (Collins 1984:v). This may suggest that each entry (i.e., a headword and its synonym list) contains all and only words that are closely related semantically. But the same synonyms appear in several lists, and headwords are themselves synonyms of other headwords, so that the lists in CT are implicitly interconnected. We seek algorithms to process all the words that are interconnected in the thesaurus into sets which share crucial semantic features.

In the first section of this paper, we characterize the properties of the CT interconnections that we discovered in our manipulation of the CT links. Because of the asymmetric and intransitive nature of these links, our main difficulty has been to devise proper means of control to keep the computed sets of words closely related in meaning. In the second section, we describe our first control measure - our manipulation of *senses* of words rather than of words themselves. In the third section, we describe automatic ways of pruning the semantic trees we obtain. In the final section, we illustrate how this work can benefit various natural language applications by providing automatic access to semantically related word senses and an automatic means for measuring semantic distance.

Properties of CT-synonyms

In the context of CT, a strong criterion for defining a set of words which share crucial semantic features is a criterion which requires every member of the set to be a synonym of every other member. The words in such a set would exhibit symmetric and transitive links. There are 27 sets of words in CT which are symmetric and transitive. Within the context of the thesaurus, these may be considered to have identical meaning. 26 out of the 27 are word pairs - the 27th is a triple - and all have a single sense and a unique part of speech.[2] These sets are given below.

allocate	=	allot	
aphorism	=	apothegm	
astonishing	=	astounding	
at_times	=	from_time_to_time	
bystander	=	eyewitness	
cemetery	=	necropolis	
congratulate	=	felicitate	
eatable	=	edible	
entomb	=	inter	
everybody	=	everyone	
exactitude	=	exactness	
greetings	=	regards	
insomnia	=	sleeplessness	
lozenge	=	pastille	
myopic	=	near-sighted	
naught	=	nought	
perk	=	perquisite	
permeable	=	porous	
piddling	=	piffling	
podium	=	rostrum	
prizefighter	=	pugilist	
prizefighting	=	pugilism	
saw	=	saying	
slattern	=	slut	
testy	=	tetchy	
triad	=	trinity	= trio
weal	=	welt	

Most of the synonymy links in CT are markedly different from these. 62% are asymmetric (e.g., *part* has *department* as a synonym, but *department* does not have *part*); and 65% are non-transitive (e.g., *part* has *piece* as a synonym; *piece* has *chunk* as a synonym; but *part* does not have *chunk* as a synonym).[3] This asymmetry and non-transitivity have been noted by others (Dewdney 1987). Thus, in order to obtain semantic sets for most of the words in the thesaurus, symmetry and transitivity are too strict. An algorithm which permits asymmetric and non-transitive links must be developed. (See Warnesson 1985 for a different approach.)

According to the substitutability definition of synonymy adopted by Collins, links should always be symmetric since if it is possible to substitute *b* for *a* in a "sensible" English context, then it is always possible to reintroduce *a*

[2] It should be noted that CT's vocabulary is limited. Thus, it does not contain the verb "perk" or the noun "saw" as an instrument of cutting. The list of transitive and symmetric sets will vary with the size of the on-line source.

[3] The percentage of non-transitive links does not include synonyms which have no entries in CT (see footnote 4); nor does it include synonyms which could not be disambiguated (see the section on sense disambiguation). Thus 65% is a conservative estimate.

into that context as a substitution for *b*. Nevertheless, we have found at least five different sources of asymmetry. 23% of the total CT-synonyms are either phrases or rare and colloquial words, which do not appear as main entries in the thesaurus, such as *dwelling place* (a synonym of *abode*) and *digs* (a synonym of *quarters*).[4] About 68%[5] of the remaining asymmetries appear to be mere oversights on the part of the lexicographers. For example, *assembly* has *throng* listed as a synonym of one of its senses, but *throng* does not list *assembly* as a synonym, although it does give *assemblage*, *congregation*, *multitude*, and other related words. Many of these omissions seem to be due to the fact that rare, very formal or metaphoric words tend not to be offered as synonyms. This may explain why *conversant*, *familiar* and *informed*, for example, are listed as synonyms of *cognizant*, while *cognizant* is not listed as their synonym. 18% are instances of hypernymy (the superordinate relation). For example, *book* lists *manual* as a synonym, but *manual* does not list *book*; instead special types of books such as *handbook* are given. This is because *book* is really a hypernym (not a synonym) of *manual*. Hypernym links are truly asymmetric in nature.

Two other sources account for the remaining asymmetries: 8% of the asymmetric cases result when a central sense of one word is synonymous with a very peripheral sense of another. One sense of *say* lists *add*, as in "He added that he would do the demonstration." The entry for *add* does not however contain this peripheral sense and deals only with the arithmetic *add*. Finally, 6% are due to vocabulary inconsistencies. For example, *record* has *annals*, *archives* and *diary* as synonyms; whereas *annals* and *archives* have the plural *records*; and *diary* has the phrase *daily record*. We believe that the CT-synonyms are nontransitive for many of these same reasons.

Is it possible to reach any noun in CT by following the synonym links to and from any other noun? The answer is NO, but almost. Computing the transitive closure over the synonyms of the noun *house*, where we include the words listed in the entry for *house* and the words whose entries list *house* as a synonym, produces a grouping containing 89% of all the nouns in CT. Obviously, with such a large number of words, it is not surprising that most bear little semantic relation to the root node.

The computational tool we have used for computing the transitive closure over synonymy is a program known as SPROUT. It was originally used (Chodorow, et al., 1985) to generate taxonomic trees from the hyponym relation as extracted from *Webster's Seventh Collegiate Dictionary* (Merriam 1963). SPROUT starts with a root node and retrieves from a designated file (in this case, a DAM file) the words that bear the given relation to the root. These words are the first-level descendents (daughters) of the root. SPROUT then applies recursively to each of the daughter nodes, generating their daughters, etc. In this way, the tree is generated in a breadth-first fashion. The process is complete when the only nodes that remain open are either terminals (i.e., nodes that have no daughters) or nodes that appear earlier in the tree, indicating a cyclic structure. The *house* tree reached closure at the 11th level.

Sense Disambiguation

Perhaps the diversity in meaning encountered in the sprout of *house* came from considering nodes to be words. Words are, of course, polysemous, so a better choice might be word senses. The CT-entry of *house* is given below. The numbers 1-6 indicate six different senses.

1. abode, building, domicile, dwelling, edifice, habitation, home, homestead, residence
2. family, household, ménage
3. ancestry, clan, dynasty, family tree, kindred, line, lineage, race, tribe
4. business, company, concern, establishment, firm, organization, outfit (*Informal), partnership
5. Commons, legislative body, parliament
6. hotel, inn, public house, tavern

[4] We had to ignore these words in our subsequent manipulation of the CT-entries because they had no synonym lists. Thus, the total number of synonyms available for processing is 221,957.

[5] The following percentages were computed on the basis of fifty random entries.

The synonyms listed for each sense are not separated into their senses. Consequently, simply following the synonyms of *house1* will not solve the problem unless each of the synonyms for it (*abode*, ..., *residence*) is marked with its appropriate sense. We have tried two automatic methods of sense marking (i.e. sense disambiguation): disambiguation by symmetry and disambiguation by intersection.

In a dictionary-style thesaurus such as CT, an entry A may have word B listed as a synonym of its nth sense, and entry B may have word A listed as a synonym of its mth sense. We can mark B in entry A as the mth sense of B, and A in entry B as the nth sense of A. An example of this type of one-to-one mapping in CT is given below.

dense (adj) 1. ... condensed ... solid ....
2. ... dull ... stupid ...

dull (adj) 1. dense .... stupid ....
2. ... callous ... unsympathetic
.
.
.
7. drab ... muted ....

Here, sense 1 of *dull* is synonymous with sense 2 of *dense*. 37% of the 287,000 synonym tokens show this type of symmetry. Of course, there are also mappings of the one-to-many variety (for example, only the first sense of *feeble* has *faint* as its synonym, whereas both senses 1 and 2 of *faint* have *feeble*), but they account for only .5% of the tokens. By this method of disambiguation-by-symmetry, we could automatically mark the senses of all synonyms in one-to-one and one-to-many relations. The third type of mapping, many-to-many, accounts for just .5% of the total, but it poses a problem for the strategy outlined above. This can best be seen by considering an example. Senses 1 and 2 of *institution* list *establishment* as a synonym, and senses 1 and 2 of *establishment* list *institution*. Is sense 1 of *institution* synonymous with sense 1 of *establishment* or with sense 2? The distribution of the terms *institution* and *establishment* cannot answer the question.

The problem of many-to-many mappings and the large percentage of asymmetric CT-synonyms led us to another method. Consider again the case of *dense* and *dull*. Evidence for linking sense 2 of *dense* with sense 1 of *dull* comes from the symmetric distribution of the two words in the entries. There is however another piece of evidence for linking sense 2 of *dense* with sense 1 of *dull*, and that is the co-occurrence of the word *stupid* in their synonym lists. Thus, the intersections of synonym lists serve as the basis for an automatic disambiguation of the many-to-many mappings, and, for that matter, for the disambiguation of the whole CT. This is similar to Lesk's suggestion for disambiguating hypernyms (Lesk 1986). The intersection method disambiguated more entries than the symmetry method, but it, too, left a certain percentage of ambiguous words. In some cases, the intersection of two words was null. For example: *successful* and *victorious* are symmetric synonyms but none of their other synonyms are shared. Their entries are given below.[6]

SUCCESSFUL:
> > 0acknowledged$ at_the_top_of_the_tree$99
best-selling$99 booming$99 efficacious$
favourable$ flourishing$0 fortunate$1.2
fruitful$3 lucky$1 lucrative$0
moneymaking$0 out_in_front$99 paying$99
profitable$1 prosperous$1 rewarding$0
thriving$0 top$ unbeaten$1 victorious$
wealthy$0

VICTORIOUS:
> > 0champion$ conquering$99 first$
prizewinning$99 successful$
triumphant$0 vanquishing$99 winning$2

In other cases, there was a tie. For example, *ripe2* has equal-size intersections with both *perfect1* and *perfect4*. In their following entries, ties are indicated by a pair of numbers joined by a period.

PERFECT:
> > 1absolute$1 complete$1.3 completed$99
consummate$2 entire$1.3 finished$2 full$1
out-and-out$ sheer$2 unadulterated$99
unalloyed$99 unmitigated$2 utter$99 whole$1
> > 4accomplished$2 adept$1 experienced$1
expert$2 finished$1 masterly$0 polished$
practised$ skilful$0 skilled$0

RIPE:
> > 2accomplished$1 complete$2 finished$
in_readiness$ perfect$1.4 prepared$1
ready$1

[6] The number following the dollar sign indicates the sense number. No number indicates that the intersection is null and therefore a sense number was not picked up. 99 indicates that the word has no entry in CT and consequently no sense numbers. 0 means that there was only one sense given in the entry.

No disambiguation resulted in either of these cases. The results obtained with each method are shown in the following table:

by symmetry:		
sense disambiguated:	103,648	(46.7%)
ties:	1,662	(0.7%)
remainder:	116,647	(52.5%)
Total number of synonyms available for processing:	221,957	
by intersection:		
sense disambiguated:	179,126	(80.7%)
ties:	6,029	(2.7%)
remainder:	36,802	(16.6%)
Total number of synonyms available for processing:	221,957	

Figure 1. Disambiguation Results

The quantitative advantage of the intersection method is evident. To determine the qualitative difference, we studied cases where the symmetry and the intersection methods conflicted. We compared fifty randomly selected entries. Of the approximately 900 synonyms listed in the entries, 337 were disambiguated by both methods. Of these, there were 33 pairs for which the two methods disagreed. 20 were symmetric ties, disambiguated by the intersection method. 5 were intersection ties, disambiguated by the symmetry method. The remaining 8 were given to two human reviewers. In 3 out of the 8, the reviewers could not determine which of the methods provided better disambiguation, as shown in the following example.

FEEBLE:
1. debilitated, delicate, doddering, effete, enervated, enfeebled, etiolated, exhausted, failing, faint, frail, infirm, languid, powerless, puny, shilpit (*Scottish), sickly, weak, weakened
2. flat, flimsy, inadequate, incompetent, indecisive, ineffective, ineffectual, inefficient, insignificant, insufficient, lame, paltry, poor, slight, tame, thin, unconvincing, weak

POOR:
1. badly off, broke (*Informal), destitute, hard up (*Informal), impecunious, impoverished, indigent, in need, in want, necessitous, needy, on one's beam-ends, on one's uppers, on the rocks, penniless, penurious, poverty-stricken, skint (*BritishSlang), stony-broke (*BritishSlang)
2. deficient, exiguous, inadequate, incomplete, insufficient, lacking, meagre, miserable, niggardly, pitiable, reduced, scanty, skimpy, slight, sparse, straitened
3. below par, faulty, feeble, inferior, low-grade, mediocre, rotten (*Informal), rubbishy, second-rate, shabby, shoddy, sorry, substandard, unsatisfactory, valueless, weak, worthless
4. bad, bare, barren, depleted, exhausted, fruitless, impoverished, infertile, sterile, unfruitful, unproductive
5. hapless, ill-fated, luckless, miserable, pathetic, pitiable, unfortunate, unhappy, unlucky, wretched
6. humble, insignificant, lowly, mean, modest, paltry, plain, trivial

The symmetry method linked *feeble2* with *poor3*, whereas the intersection method linked *feeble2* with *poor2*. The remaining 4 cases were somewhat clearer. In 3, the intersection method performed better; in one, the symmetry method was superior. To conclude, the best disambiguation algorithm would be a combination of the two methods. We are currently studying more cases where the methods disagree in order to determine how they should be combined. In the following, though, we rely on disambiguation by intersection.

Transitivity

After numbering each token in CT by sense and disambiguating senses, we sprouted from the first sense of *house*. Each node in the new sprout is not a word anymore, but a specific (numbered) sense of a word. Words not disambiguated by the intersection method were ignored in the new sprout. Sense disambiguation did not significantly improve the results of the sprout. The sprouting closure of *house1* contains 85% of the total number of noun senses in CT.

Using senses instead of words, we recomputed the number of sets which are symmetric and transitive (see section on CT-properties above) and found 86. Given below are some of the new sets.[7]

adhesive$2	= glue$1	= paste$1
beak$1	= bill$5	
conservatory$0	= hothouse$1	
draw$11	= tie$7	
grade$3	= gradient$0	
grouch$2	= grouse$2	= grumble$3

[7] As before, sense numbers follow the dollar sign.

myopic$0 = near-sighted$0 = short-sighted$0
poison$1 = venom$1
spectator$0 = witness$1
well-off$2 = well-to-do$0
wolf$2 = womanizer$0

Why is sense disambiguation so ineffective in restricting the number of nodes encountered in sprouting? Consider for example the thesaurus separation into senses for *building*:

1. domicile, dwelling, edifice, fabric, house, pile, structure
2. architecture, construction, erection, fabricating, raising

Sense 2 is a mixture of the act of building, the object built, and its design. This indicates that poor sense separation in CT is responsible for spuriously related word senses. We feel that a reliable indication of poor sense separation in CT might be an intersection of two synonyms which is of the size 1. For example, the intersection of *building2* and *erection1* contains only *construction.*

Erection:
1. assembly, building, construction, creation, elevation, establishment, fabrication, manufacture
2. building, construction, edifice, pile, structure

By ignoring CT links with intersections of size 1 we were able to eliminate some of the problematic senses and reduce the sprout to include only 76% of the total CT-nouns, as opposed to the previous 85%.

In an attempt to maintain semantic content, we have explored automatically pruning the sprout tree when a semantically irrelevant branch is generated. Before any CT-synonym is accepted as a node of the tree, its descendents are checked against the immediate descendents of the root node. If their intersection is not null, the node is accepted into the sprout tree. We have experimented with a few variations: choosing either the daughters or both daughters and granddaughters of either the root node or the branch node. We have also varied the size of the intersection. A promising scheme involves checking the daughters of each node against the daughters and granddaughters of the root, discarding nodes whose intersection is of size 1. When pruned this way, the sprout tree of *house1* reached transitive closure with a total of 173 noun senses, which constitute 1.4% of the total noun senses in CT. Closure was reached at the 4th level. The first following list includes most of the nodes that were rejected by the pruning method. The second list includes most of the nodes that were accepted.[8]

2-home$3 2-fabric$2 3-barracks$0
3-assembly$2 3-composition$1 3-erection$1
3-fabrication$1 3-figure$3 3-form$7
3-formation$2 3-shape$1
3-design$4 3-make$12 3-making$1
3-manufacture$3 3-mould$2 3-organization$1
3-production$1 3-house$2 3-point$2
3-orientation$1 3-quarter$1 4-chamber$1
4-framework$0 4-system$1 4-anatomy$2
4-build$4 4-hull$1 4-physique$0
4-rack$1 4-skeleton$0 4-arrangement$1
4-configuration$0 4-format$0
4-organization$2 4-architecture$2
4-turn$17 4-conformation$0
4-constitution$2 4-method$2
...
4-entourage$2 4-field$3 4-aspect$2

0-house$1 1-abode$0 1-building$1
1-domicile$0 1-dwelling$0 1-edifice$0
1-habitation$1 1-home$1 1-residence$1
1-address$1 1-establishment$4 1-place$5
1-seat$4 2-lodging$0 2-quarters$0
2-lodgings$0 2-mansion$0 2-pile$4
2-structure$2 2-construction$1
2-erection$2 2-household$1 2-pad$4
2-location$0 2-situation$1
2-whereabouts$0 3-accommodation$2
3-billet$1 3-apartment$0 3-frame$4
3-make-up$2 3-structure$1 3-bearings$0
3-locale$0 3-place$1 3-position$1
3-site$1 3-spot$2 3-emplacement$1
3-locality$2 3-seat$2 3-setting$0
3-station$1 3-environment$0 3-scene$2

Applications

One way in which sprout trees of synonyms may prove to be useful is in measuring the semantic distance between words. It is possible, for example, to sprout from two different root nodes until their trees intersect, that is, until they have a common node, which, with further sprouting, will become a common branch. We believe that a common node indicates a common semantic aspect and that an algorithm for measuring semantic distance between words can be formulated on the basis of the common nodes of their trees. Intuitively, the algorithm will depend on the number of

[8] The number preceding the word indicates the level on which it was encountered in the tree. The number following the dollar sign indicates its sense number.

common branches and on the level at which they occur in the respective trees. Here we are taking a somewhat simple view in considering only the first common node encountered. Our SYNCHAIN program produces simultaneously two sprout trees from two root nodes. After all words on a level are encountered in the two trees, the program checks whether the trees intersect. It stops when a common node is encountered. The user specifies the two root nodes and the level to which the trees should be sprouted. The program provides a common node, if one was encountered. This is illustrated in the two following examples:

```
synchain apartment$0 house$1 3
apartment$0 -> place$5 <- house$1

synchain apartment$0 suit$5 3
chain could not be constructed
```

In the first example, the *apartment$0* tree and the *house$1* tree intersect on their first level. Both have *place$5* as their daughter. In the second example, the *apartment$0* tree and the *suit$5* tree (in the meaning of garment) do not intersect as far as the third level. This suggests that the word senses of the first pair are much closer in meaning than those of the second.

This distinction can assist in the analysis of natural language text (for purposes of translation, text critiquing and others), by providing semantic information to a syntactic parser. In particular, we have in mind a parser such as the PLNLP English Parser, PEG (Jensen 1986), which cannot resolve cases of syntactic ambiguity on its own. Consider the following pair of sentences with their PLNLP analyses:

```
the man visited the apartment in the house.
DECL NP     DET     ADJ*  "the"
            NOUN*         "man"
     VERB*                "visited"
     NP     DET     ADJ*  "the"
            NOUN*         "apartment"
     ?      PP      PREP  "in"
                    DET   ADJ*  "the"
                    NOUN*       "house"
     PUNC "."
```

Figure 2. Parse Tree 1

```
the man visited the apartment in a white suit.
DECL NP     DET     ADJ*  "the"
            NOUN*         "man"
     VERB1*               "visited"
     NP     DET     ADJ*  "the"
            NOUN*         "apartment"
     ?      PP      PREP  "in"
                    DET   ADJ*  "a"
                    AJP   ADJ*  "white"
                    NOUN*       "suit"
     PUNC "."
```

Figure 3. Parse Tree 2

PEG produces similar analyses for the two sentences, where the prepositional phrases are attached to the closest head, that is, to the noun *apartment*. A question mark indicates an alternate attachment to the verb *visited*. Semantic information is needed to resolve the attachment ambiguity (Jensen and Binot 1986). Our measure of semantic distance can determine the proper attachment in this case. If the two nouns are semantically close, the attachment displayed by PEG is the more plausible one. If the two nouns are semantically distant, the alternate attachment is more plausible.

An automatic measure of semantic distance can assist information retrieval systems as well. One can conceive of a system which will retrieve documents containing synonyms of the key word by first searching for a very restrictive set of synonyms (first-level synonyms perhaps). If not enough documents are retrieved, words that are more distant semantically can be searched for as well. Another application for which a sprouted synonym tree is useful is third-generation on-line dictionary systems (Neff, et al., 1988). Among other things, these systems display synonyms to users who are editing natural language texts. The list of synonyms presented by the system can be arranged according to the semantic distance between the word interrogated and the words on the synonym list. It should be noted, however, that for this application, words need to be arranged according to additional parameters as well. Synonyms that are polysemous or rare may be poor substitution candidates in a general text.

Finally, we are now investigating the possible use of our tools by lexicographers who wish to update and revise an existing on-line thesaurus. Easy access to asymmetric links, to synonyms with very small intersecting lists, to lists of words that are pruned from sprout trees, and to any other sorted information that we can provide automatically should make the

work of lexicographers much more easily manageable. Our goal is to develop a tool that will automatically locate all different types of inconsistencies and oversights in the thesaurus (Ravin, et al., in preparation).

Conclusion

We have explored the nature of the implicit synonym links in CT and have found it complex but promisingly rich. Our goal is to continue to improve the automatic extraction of information from this source, until we form acceptable sets of semantically related words. These sets will have to satisfy both human intuitions about meaning and some more theoretic linguistic criteria. To these sets, we will add information from other on-line sources. This direction of research seems promising as a first step towards the automatic organization of meaning.

REFERENCES

Byrd. R. J. (1986), "Dictionary Systems for Office Practice," *Proceedings of the Grosseto Workshop "On Automating the Lexicon"*. Also available as IBM Research Report RC 11872.

Byrd, R. J., G Neumann, and K. S. B. Andersson (1986) "DAM - A Dictionary Access Method," IBM Research Report.

Chodorow, M. S., R. J. Byrd, and G. E. Heidorn (1985) "Extracting Semantic Hierarchies from a Large On-line Dictionary," *Proceedings of the Association for Computational Linguistics*, 299-304.

Collins (1984) *The New Collins Thesaurus*, Collins Publishers, Glasgow.

Dewdney, A. K. (1987) "Word ladders and a tower of Babel lead to computational heights defying assault," *Scientific American*, August 1987, 108-111.

Grishman, R. (1986) *Computational Linguistics*, Cambridge University Press, Cambridge.

Jensen, Karen (1986) "PEG 1986: A Broad-coverage Computational Syntax of English," Unpublished paper.

Jensen, Karen and Jean-Louis Binot (1987) "Disambiguating Prepositional Phrase Attachments by Using On-Line Dictionary Definitions," to appear in *Computational Linguistics*, special issue on the lexicon. Also available as IBM Research Report RC 12148.

Katz, J. and J. Fodor (1963) "The Structure of a Semantic Theory," *Language*, 34, 2:170-210

Lesk, M. (1986) "Automatic sense disambiguation using machine-readable dictionaries: How to tell a pine cone from an ice cream cone," *Proceedings of 1986 SIGDOC Conference*, Canada.

Marcotorchino, F. (1986) "Maximal Association Theory", in *Classification as a Tool for Research*, W. Gaul and M. Schader, eds., North Holland.

Merriam (1963) *Webster's Seventh New Collegiate Dictionary*, G. & C. Merriam, Springfield, Massachusetts.

Neff, M. S., R. J. Byrd, and O. A. Rizk (1988), "Creating and Querying Lexical Data Bases," *ACL Second Conference on Applied Natural Language Processing.*

Quine, W. (1960) *Word and Object*, MIT Press, Cambridge, Massachusetts.

Ravin, Y., M. Chodorow, and H. Sachar (in preparation) "Tools for lexicographers revising an on-line thesaurus."

Warnesson, I. (1985) "Optimization of Semantic Relations by Data Aggregation Techniques," *Journal of Applied Stochastic Models and Data Analysis*, 1,2 (December), J. Wiley, New York.