Computers Math. Applic. Vol. 23, No. 6–9, pp. 391–402, 1992
Printed in Great Britain.

0097-4943/92 $5.00 + 0.00

SENSE AND PREFERENCE

BRIAN M. SLATOR
The Institute for the Learning Sciences
Northwestern University, Evanston, IL 60201, U.S.A.

Abstract—Semantic networks have shown considerable utility as a knowledge representation for Natural Language Processing (NLP). This paper describes a system for automatically deriving network structures from machine-readable dictionary text. This strategy helps to solve the problem of vocabulary acquisition for large-scale parsing systems, but also introduces an extra level of difficulty in terms of word-sense ambiguity. A Preference Semantics parsing system that operates over this network is discussed, in particular as regards its mechanism for using the network for lexical selection.

1. INTRODUCTION

Many of the problems of Artificial Intelligence (AI) revolve around issues of knowledge, its representation, and its acquisition. Indeed, it is widely accepted that large and consistent sources of knowledge must be in place before any serious attempt at implementing computational intelligence can begin. Semantic networks, and equivalent frame-based representation schemes, have received a great deal of support over the years, both through psychological studies of human memory, and through surviving the test of hard use as the basis of AI systems.

The need for prior knowledge is nowhere more apparent than it is in Natural Language Processing (NLP) research. Natural language expression is a fascinating mixture of redundancies on one hand and unstated assumptions on the other. In either case, knowledge of the world is invariably necessary, and knowledge of particular domains almost always required, before a system of language analysis can determine what is being written or spoken about. Given these observations, two interrelated points of interest arise: how this knowledge is to be gathered to begin with, and how it is to be best represented for the purposes of NLP.

2. KNOWLEDGE ACQUISITION FROM TEXT

Knowledge structures, network or otherwise, can be gathered from text; this is a widely held and non-controversial assumption. The intuition at work is simply that much knowledge is stored in text, and students of the world, as we all are, are generally able to learn from text. For AI, knowledge from text is an attractive methodology in any case, since it means transforming existing knowledge rather than somehow inventing it wholesale (cf. the 1990 AAAI Symposium on Text-based Intelligent Systems), and it makes particular sense when the domain of interest is relatively general and the demands on the knowledge base are likely to be extensive.

A system for extracting structured lexical semantic information from machine-readable dictionary text, focusing in particular on the network structures that result, is described below. These structures are useful for the machine analysis of natural language, particularly for word sense disambiguation, because the source material is itself large and consistent, being the fruit of many work-years of lexicography. These structures form the basis of an analysis system, based on the theory of Preference Semantics, which is also described below.

This work was supported in part by ASEND/EPSCoR Grant 4248-3341, administered by the North Dakota Board of Higher Education and by the New Mexico State University Computing Research Laboratory through NSF Grant No. IRI-8811108 to Dr. Yorick Wilks.

3. DICTIONARY STRUCTURE

Network structures in NLP serve two complementary purposes. First, the meaning of a lexical item reduces to the other items it connects to, and the relations that hold between these items. It is a truism that words are defined in terms of other words, and it is an operational fact of life that this connectivity, and the attendant notions of spreading activation and semantic distance across a network, are the devices that we use to compute the meaning of a lexical item in a domain of discourse. Second, networks and the relations they capture are the bread-and-butter of logical inference. In large measure, inferencing amounts to traveling around a network for the purpose of discovering generalizations and exploiting properties like inheritance and transitivity. Network structures from dictionary text are formed from both explicit and implicit dictionary structures.

3.1. Explicit Dictionary Structure

The Longman Dictionary of Contemporary English (LDOCE) [1], is explicitly organized according to a variety of structural principles. LDOCE is a full-sized dictionary designed for learners of English as a second language which contains 41,122 headword entries, defined in terms of 72,177 word senses, in machine-readable form (a type-setting tape). The book and tape versions of LDOCE both employ a network hierarchy of grammatical codes of about 110 syntactic categories which vary in generality from, for example, *noun* to *noun/count* to *noun/count/followed-by-infinitive-with-TO*. The machine readable version of LDOCE also contains "box" codes (also called TYPE codes) and "subject" codes (also called PRAGMATIC codes) that are not found in the book. The TYPE codes are comprised of a set of primitives such as "T" (for *abstract*), "C" (for *concrete*), and "A" (for *animate*), organized into a type hierarchy. This hierarchy of primitive types conforms to the classical notion of the IS-A relation as describing proper subsets, except that Longman has introduced additional "composite" type codes, such as "E" (for *solid+liquid*, which is used on such disparate lexical entries as the nouns "bouillabaisse—a strong-tasting dish made from fish," and "magma—hot melted rock found below the solid surface of the earth," and the adjective "treacly—(of a drink or liquid food) too thick and sweet"). These added semantic types transform the standard Aristotelean hierarchy into a network of near-primitives; however, the important property of quantifiable semantic distance is preserved, and this proves to be extremely useful for NLP purposes. These primitives codes are used in LDOCE to assign type restrictions to nouns, and to mark selection restrictions on the arguments of verbs and adjectives.

The PRAGMATIC codes in LDOCE are another set of terms organized into a hierarchy. This hierarchy consists of main headings such as **engineering** with subheadings like **civil** and **electrical**. These terms are used to classify words by subject area. For example, one sense of "current" is classified as **geology-and-geography** while another sense is marked **engineering/electrical**. The LDOCE pragmatic coding system divides the world up into 124 major subject categories ranging from **aeronautics**, **aerospace**, and **agriculture**, through **glass**, and **golf**, to **vehicles**, **water-** and **winter-sports**, and **zoology**. Many of these subjects are further subcategorized (for example, under **agriculture** are **soil-science** and **horticulture**, and under **zoology** are **entomology**, **ornithology**, and **ichthyology**), so there is a total of 369 different subject codes in the LDOCE pragmatic system. However, the LDOCE pragmatic hierarchy as given is flat (only 2 layers deep), and the 124 major categories have equal and unrelated status; for example, **business** and **economics** are both at the top of the tree and are unconnected, the same is true of **science** and **zoology**.

In order to make this LDOCE pragmatic coding system more useful for meaning and inference, a deeper structure has been imposed onto the LDOCE pragmatic world [2], in order to, for example, make explicit that words classified under **botany** have pragmatic connections to words classified as **plant-names**, as well as connections with other words classified under **science** (connections not made by the LDOCE pragmatic hierarchy as given), and that these connections are useful to exploit when attempting to determine the subject matter of a text, or when attempting to choose the correct sense of polysemous words [3]. The original LDOCE scheme, with 124 top-level elements and 245 second-level elements in a two-level structure, has been transformed into a structure that is 6 levels deep in some places, with only 7 top-level elements (in the present system

these are **communication, economics, entertainment, household, politics, science,** and **transportation**).

3.2. Implicit Dictionary Structure

Dictionary definitions are often characterized as being composed of *genus* and *differentia* terms (e.g., [4]). The genus term is an "upward-pointing" reference (sometimes called a hypernym), that places a word within a superordinate class; they distinguish between members of a primitive TYPE. For example "ammeter" and "automobile" are both of TYPE *solid*, but the genus of the first is "instrument" while the genus of the second is "vehicle." These genus terms form an IS-A network of word senses that captures the conceptual structure of the language.

The difficulty in automatically deriving a network of word senses from a dictionary is that definition texts are not typically sense-tagged, and so while the genus of "ammeter" is the word "instrument", it is nowhere explicitly stated that this refers to "instrument1—instrument as tool" rather than "instrument2—a musical instrument." Of course, if the IS-A network of *genera* is to be meaningful, this distinction is exactly what is needed; inferences about the use of "ammeter" in text depend on knowing the proper sense of its genus.

This problem was originally tackled by Amsler [5,6], with the Merriam Webster Pocket Dictionary [7], where paid human "disambiguators" sense-tagged the words in a definition which "made a significant semantic contribution to an IS-A link" [6, p. 55] with the headword being defined. This resulted in a "tangled hierarchy" of word senses to represent the structure of the dictionary, and also revealed certain difficulties to do with the consistency of these networks in the face of a significant number of problem definitions. The problems stem from cases where the head of the first noun phrase (the usual place to find a genus term) appears vacuous, and another word in the definition gives the relevant information about the headword. Amsler and White [6], kept a list of these words, referring to them as partives and collectives; Nakamura and Nagao [8] call them Function Nouns; Chodorow *et al.* [9], using Webster's Seventh [10], refer to a subset of these as "empty heads."

A recent analysis by Guthrie *et al.* [11] takes issue with these earlier characterizations by observing that the Amsler & White handling of the phenomena is unhelpful in many cases, that the Nakamura & Nagao system collapses certain cases that should be kept distinct, and that the Chodorow *et al.* solution treats entities as vacuous when indeed they are not. Guthrie *et al.* identify a set of these problem cases, referring to them as "disturbed heads." They go on to give an algorithm for automatically finding the correct sense for the genera of noun definitions, including those with "disturbed heads." This system takes the output of the Lexicon Provider [12-17], which produces definition parse trees and genus terms from them, and passes it to the Genus Disambiguator program, along with the relevant TYPE and PRAGMATIC codes from LDOCE. The Genus Disambiguator selects the correct genus sense by matching codes, and through weighing alternatives, in terms of semantic distance and pragmatic coherence in the explicit LDOCE networks described above. The result is another network structure of IS-A links, but one where word senses are connected with disambiguated word senses rather than primitive TYPE labels.

The picture of dictionary structure that emerges, then, is one where word senses are multiply connected to each other, and to other entities, through a variety of link types (see Figure 2, below). The structure is valuable in terms of its applications to NLP, and it is automatically derived from existing text sources rather than by hand.

3.3. Parsing Dictionary Entries

The first step in discovering implicit dictionary structure involves an analysis of the definition text of an entry, to capture the form of the entry in some meaningful way. In this system, that is accomplished by parsing the entry into its constituent structure, and then searching for and labelling meaningful elements within that structure.

To do this, a syntactic parser and pattern-matcher have been implemented. The parser operates over the LDOCE controlled vocabulary of 2000+ root forms by applying a grammar of conventional phrase-structure rules. The noun definitions of LDOCE invariably begin with a

noun phrase (NP) which is often followed by prepositional phrases and/or relative clauses (the exceptions are those noun entries defined in terms of a synonym, and these are usually in the form of an explicit cross-reference to another item defined in LDOCE).

The parser returns one or more phrase-structure trees—usually more, since almost all LDOCE definitions are in some way ambiguous. Heuristic procedures then select one from this set of trees; the preference is for trees whose leading constituent's head matches the grammatical category of the headword (the word being defined, whose definition text is under analysis), and for trees made up of "longest strings", and hence the fewest constituents.

This parser is very successful at discovering the proper syntactic structure for the leading parts of content word definitions (where the genus terms are found), and it returns a correct analysis in over 99% of the cases. The latter parts of definitions sometimes go astray, and occasionally fail completely, and these cases are the subject of ongoing work. In the case of parsing failure, where no grammar rules match and the processing halts, a partial parse is returned by appeal to the chart data structure that the parser constructs (see [18] for a good review of chart parsing).

```
(ammeter
  (NP
    (DET . an)
    (N . instrument))
  (PP
    (PREP NP)
    for
    (NP
      (N . measuring)))
  (COM . ,)
   (PP
     (PREP NP)
     in
     (NP
       (N . amperes)))
   (COM . ,)
    (NP
      (DET . the)
      (N . strength))
    (PP
      (PREP NP)
      of
      (NP
        (DET . an)
        (ADJ . electric)
        (N . current))))
```

Figure 1. Phrase-Structure Tree for "ammeter" from the LDOCE Chart Parser.

The parse tree for *ammeter*, defined in LDOCE as *an instrument for measuring, in AMPEREs, the strength of an electric current* ... appears (in a cleaned up form) in Figure 1. It should be pointed out, however, that this parser is for the definitions of content words defined in LDOCE—this is not a parser for general English.

A recent enhancement to this system has involved the results of a compositional-reduction method developed by [19]. It was found that a total of 3,860 word senses, of the 2,000+ controlled vocabulary words, make up the defining senses of LDOCE (the word senses that are actually used in the definitions of other members of the controlled vocabulary). About half of these 2,000 words (1051 words) have single defining senses, about a quarter have two defining senses, and another quarter have multiple defining senses. In the straightforward case, then, if a word is a member of the list of 1051, and if it is found in the definition of a controlled vocabulary word, that word can be unambiguously sense-tagged without futher processing.

The dictionary parser category tags the words in each definition (as to noun, verb, etc.). The vocabulary of the parser has been augmented by the defining sense lists; therefore, if a word is a member of the defining vocabulary used in only a single sense in the definitions of the controlled vocabulary (i.e., in the list of 1051), and if that word is found in the definition of a controlled vocabulary word, then that word is unambiguously sense-tagged. This prior knowledge of the defining vocabulary of LDOCE reduces the search space of the parser. Further, the parser will soon be extended to have the ability to sense tag other words (those with two or more defining senses), since there are cases where choosing the correct category (which the parser does) will effectively choose the correct sense.

This process identifies an inventory of so-called primitive senses in LDOCE, and from this it is possible to unambiguously sense tag some of the words in the definitions of the controlled vocabulary; but this method does not directly apply to the majority of the dictionary, which is either outside the controlled vocabulary or ambiguous with respect to the definitions of the controlled vocabulary words.

3.4. IS-A Links from Parse Trees

Once the parse tree for a definition text has been chosen, several things are possible. The first thing is to find the word or words in the definition that make up the genus of the headword, since these form an IS-A network. The genus of a definition, in the usual case, is a more general word that identifies some sort of superordinate class for the headword: to repeat an example, an "ammeter" IS "an instrument ... ."

There are difficulties and special cases involved in choosing the genus word, but the issues here are principally syntactic. More difficult is the problem of choosing the correct sense of the genus word, which requires a different kind of analysis.

The Genus Disambiguator (GD) [11] looks at the headword, and the semantic and pragmatic codes associated with it, and matches these against the codes associated with each sense of the genus term, as defined in LDOCE. The GD does a good job of choosing genus senses by following the following strategy:

1. choose the genus sense whose semantic codes identically match with the headword, if possible;
2. if not, choose the sense whose semantic category is the closest ancestor to the semantic category of the headword;
3. in the case of a tie, the subject codes are used to determine the winner;
4. if subject codes cannot be used to break the tie, the first one of the tied senses which appears in the dictionary is chosen (since more frequently used senses are listed first in LDOCE).

For example, suppose the following definition were under examination:

flute -
a pipelike wooden or metal musical instrument with finger holes ... (the codes in LDOCE associated with "flute" are **J** for "movable-solid" and **MU** for "Music")

The genus of **flute** is the word "instrument;" therefore, the input to the Genus Disambiguator is the list:

(flute, movable-solid, Music, "instrument")

The following are the first two LDOCE definitions for "instrument."

instrument-1 -
an object used to help in work: medical instruments (the codes in LDOCE are "movable-solid" and HWZT for "Hardware/Tools")

instrument-2 -
... an object which is played to give musical sounds (such as a piano, a horn, etc.) ... (codes are "movable-solid" and "Music").

In this case both the first and second senses of **instrument** are marked as "movable-solid", which matches perfectly with the semantic codes for **flute**. However, the tie is broken by appeal

to the subject code, Music, which selects the second sense of **instrument** as the genus of **flute**, and the output is "instrument-2." Hence, an IS-A link is added to the network of word senses to connect **flute-1** and **instrument-2**.

3.5. Other Relations from Parse Trees

There are other meaningful relations that can be gathered from the text of dictionary definitions. The most tractable is connected with the "disturbed head" phenomena discussed in [11], where certain genus terms, mainly those that serve some sort of "collective" function in a definition text, can be processed into a network containing IS-A links, HAS-MEMBER links, and MEMBER-OF links.

For example, consider the following LDOCE definition:

canteen -
(British English) *a set of knives, forks and spoons, usu. for 6 or 12 people*

Since the genus term, "set", is a collective (as are other words like "group", "class", etc.), the usual Genus Disambiguator (GD) processing will reliably choose the correct sense of "set" as the genus of "canteen." Further, it is reasonable to construct a HAS-MEMBER link from "canteen" to each of "knives", "forks" and "spoons" (and a MEMBER-OF link going back the other way). In addition, it is probable that the correct sense of the HAS-MEMBER elements can be determined by a code matching algorithm of the GD type, but this hypothesis has not been fully tested at this point.

Another source of information in dictionary definitions is discovered through locating what are often referred to as "defining formulas" [20], which attempts to identify recurring phrasal patterns in definitions, and then seeks to extract selectional restrictions and co-occurrence relations between words being defined and the words in the definitions. For example, in the definition for "ammeter", which is *an instrument for measuring, in AMPERES, the strength of an electric current* ... it seems clear that the purpose of an "ammeter" is *for measuring*.

When patterns like this are found in a definition it is reasonable to create a link from "ammeter" to "measuring" that is labelled PURPOSE. There are several such patterns that occur regularly in LDOCE definitions. However, the process of assigning sense tags to these terms is more problematical than the Genus Disambiguator case (unless the parser enhancement, exploiting the "defining vocabulary" discussed above, is able to sense tag the relevant words in the parse tree). These methods of creating new links in a network constitute an area of open research.

4. NATURAL LANGUAGE UNDERSTANDING

There are a great many problems involved with understanding natural language with a computer. These problems range from high-level difficulties such as ascribing beliefs to multiple agents in discourse, through the well-known problems of syntactic analysis and semantic interpretation, to the low-level problems of reliably decoding morphological derivation and even to the seemingly basic question of finding sentence and word boundaries in text. Other questions, like the role of context in the meaning of expressions, or the nature of and handling of ill-formedness, seem to cut across all the levels.

It is mainly the case that, in order to get working systems, certain of these NLP problems are pushed to the side, ignored, or otherwise finessed, in order to get at some tractable sub-problem domain. For example, the problem of morphological analysis can be circumvented by limiting the domain of interest to a certain body of texts to be analyzed, and then creating a lexicon of word representations that includes all the spelling forms in the corpus. In this way, morphology can be coded into the lexicon, rather than computed; to show, for example, that the root form of "displaced" is "place", but the root form of "displayed" is "display." This strategy saves against the effort of programming a general solution to the special cases of the morphology problem (which is not, as is sometimes claimed, in any way a "solved" problem).

One of the most difficult problems to finesse in NLP is the problem of word sense selection. Word sense ambiguity ranks with the more famous problem of structural ambiguity as being

among those which are practically impossible to get around. Consider the following sentences.

(S1) He measured the current with the ammeter on his workbench.

(S2) He measured the current with the ammeter on his coffee break.

It is most useful to think of parsing these sentences into a case-frame representation and then imagining these frames as separate knowledge bases for a question-answering system. The case-frames are approximately these:

(K1) measure:
AGENT: he
OBJECT: current
INSTRUMENT: ammeter
LOCATION: workbench

(K2) measure:
AGENT: he
OBJECT: current
INSTRUMENT: ammeter
TIME: coffee break

The indentation in K1 indicates that the location of the ammeter is somewhere on the workbench, while in K2 the time of the measuring is during the coffee break. That is, the "on" preposition in the first sentence modifies the INSTRUMENT of the verb, but the "on" preposition in the second sentence indicates the TIME of the verb itself. Stated another way, the "ammeter" is "on the workbench", but the "measuring" is "on the coffee break." These facts about the state of the world (where, for present purposes, S1 describes a world and S2 describes another), are quite obvious to most human readers. However, sorting out where each prepositional argument belongs, and deciding which semantic relation each one marks, is what structural ambiguity is all about.

Resolving the structural ambiguity allows question-answering systems to correctly reply to the question "Where is the ammeter?" in the context of knowledgebase K1, and to correctly reply to "When was the measuring?" in the context of knowledgebase K2 (but not, of course, the other way around). However, and this is the core of the word-sense ambiguity problem, neither of the knowledge bases can supply the answer to the question "What is current?". Why? Because the lexical entry for "current", which is presumably accessible through the lexicon for the system in order to answer questions of this type, might only record that "current" is a noun. Or, if there is a machine-readable dictionary on hand for handling this type of query, it will quite reasonably return the first noun sense of "current", which looks like this:

current 1 —
a continuously moving mass of liquid or gas, esp. one flowing through slower-moving liquid or gas: *The current is strongest in the middle of the river.* — *air currents*

And, of course, this is wrong. The correct answer to the question, "What is current?", in the context of either knowledgebase K1 or K2, is the definition of the second noun sense of "current" which is:

current 2 —
the flow of electricity past a fixed point -see also ALTERNATING CURRENT, DIRECT CURRENT

And therein lies the rub: the question-answering system cannot make this distinction and correctly answer the question, unless the parsing system has already selected the correct word-sense for "current." How can this be achieved?

5. PREFERENCE SEMANTICS

Preference Semantics [21–24] is a theory of language which claims that all languages are to be understood and generated by means of semantic as well as syntactic coherence, and which

holds that there is an inference-based set of principles that explain normal language use and that these principles extend to the resolution of non-standard usage and metaphor. In these terms, the resolution of structural ambiguity (the attachment problem) and the resolution of word sense (lexical) ambiguity, are both handled as a matter of discovering and evaluating the various semantic preferences that are satisfied and broken under competing interpretations, and choosing the one that is best. Usually this means the interpretation that is the most semantically "dense," where density is a function of meaningful coherence.

5.1. Ambiguity Resolution

The semantic role of "ammeter" in both sentences, S1 and S2 above, is INSTRUMENT because the verb "to measure" prefers to have an argument of that type, and because the phrase "*with an ammeter*" is an entity that can satisfy that role; and this configuration is more strongly preferred than one where, say, the ACCOMPANIMENT role is considered—as in the case of "a man ate a meal *with a friend*."

Similarly, the correct sense of "current" in both sentences S1 and S2 is current2, and this is resolved because of its greater coherence with the representation of "ammeter." Both "ammeter" and "current2" are defined with LDOCE PRAGMATIC codes to be in the **engineering/electrical** subject area, as opposed to "current1", listed above, which is marked with PRAGMATIC codes as being part of the **geology-and-geography** subject area.

Other problems with these examples all center around the attachment of the "on" prepositional phrases. This is pursued further, below.

5.2. Sense Selection: Correct vs. Best

Up to this point, word sense selection has been discussed in terms of "correctness"—the implication being that lexical choices are discrete and quantifiable in some way, and that there is one right, and some other wrong word senses to choose among. This is an over-simplification.

This can be seen by again posing the question "What is current?" in the context of sentence S1 or S2. Clearly, the 2nd sense of "current", about "the flow of electricity", is better than the 1st sense, about the "moving mass of liquid or gas." But the 3rd sense of "current" is also a viable candidate, since it is defined as follows:

current 3 — *tech* the rate of flow measured in AMPEREs

Now the question arises, "which is better, current2 or current3?" It can be seen that both are superior to "current1," but the choice between the two "good" candidates is not at all straightforward.

The answer to this conundrum, as is so often the case in NLP, lies outside the province of the system. In other words, this particular question of lexical selection depends on the context of the question, and not on the statements being parsed for the purpose of natural language understanding. As often happens in NLP, the answer to a question is at a different level than the question itself. In this case, there is no "correct" choice between "current2" and "current3" that can be made at the level of lexical selection; rather, this is an example of a problem, mentioned in the section on natural language understanding, above, that seems to cut across the entire arena of natural language understanding.

The best that can be hoped for in situations like this, is that the system will choose what seems to be the "best" among the alternatives, and hope that it is good enough and not clearly wrong. This strategy, incidentally, is one of the operating assumptions of Preference Semantics.

5.3. The Preference Machine

PREMO: the **PRE**ferene Machine Organization [25,26] is a knowledge-based parser for text, based on the Preference Semantics theory of language. PREMO operates over the collection of lexical items produced by the Lexicon Provider, described above, which is, equivalently, the network of word senses and primitive elements described in Section 3 on dictionary structure.

Parsing in PREMO is a matter of moving left-to-right through each sentence in a text, assimilating each word (and its lexical representation) into whatever pre-existing structure has been

constructed. Word and phrasal elements are organized according to a set of phrase structure rules, where patterns of existing structure and lexical elements are associated with structure building operations.

After every step in the process the resulting structures are evaluated according to a metric which takes into account the semantic preferences and grammatical predictions that are encoded within the structures. Competing parses are ranked according to this metric, and this determines which of the possible partial parses are pursued. There may be several competing parsings in play at any given moment, since every opportunity for structural or lexical ambiguity is a choice point in the space of possible parses.

The PREMO control mechanism is a priority queue of these competing partial parse structures. The operational metaphor at work is that of the standard operating system model, where each partial parse is captured in a "process control block" that is assigned a "time slice" that allows it to move one word forward in the text being parsed. After each time slice the "priority" of the process is re-computed, and it either gets another time slice or finds itself relegated to a position somewhere back in the priority queue.

5.4. Economies of Scale

A parsing system which operates over real world text is faced with a large and hard problem because, as outlined above, the search space is combinatorially large with branchings at every choice point for structural and lexical ambiguity. Consider, for example, sentence S1 from above.

(S1) He measured the current with the ammeter on his workbench.

This simple-seeming sentence is composed of 9 unique words. Of these, LDOCE defines multiple senses for 7 of them (only "ammeter" and "workbench" are defined in terms of a single sense): "He" has 5 senses (3 pronoun and 2 noun), "measure" has 26 senses (15 nouns, 7 of these idiomatic uses such as "measure one's wits", 7 verbs, 4 of them idiomatic, plus 4 more phrasal senses like "measure against" and "measure up"), "current" has 7 senses (3 adjective, 4 noun), the prepositions "with" and "on" have 20 and 19 senses respectively (in addition to which "on" has 14 adverbial senses and 4 adjective senses), "his" has 4 senses (2 determiner and 2 pronoun), and even "the" is defined in LDOCE in terms of 18 determiner senses and 3 adverbial senses (as in "he has the greatest difficulty with ... ").

The combinatorics are obvious and awesome. Leaving aside the inflated difficulty of "the," the lexical ambiguity of sentence S1 makes for a total of

$$5 \times 26 \times 7 \times 20 \times 37 \times 4 = 2,693,600$$

different readings for sentence S1!

And this figure ignores the fact that sentence S1 is also structurally ambiguous (is it the "measuring" that is "on the workbench," or is it the "ammeter" that is "on the workbench", or, more implausibly, is it the "current" that is "on the workbench"). The structural difficulty multiplies the level of ambiguity by at least 2.

Of course, these figures are quite easily reduced by parsing, or even by pre-processing: for example, in sentence S1 only the word "measure" is defined in a verbal sense, therefore it MUST be the verb in this particular sentence, and this alone reduces the combinatorics from 2,693,600 to 1,139,600, a factor of 58%. On the other hand, these calculations point to an inherent difficulty with NLP for text. When operating over large texts, and when using machine lexicons to tackle the vocabulary problem, the problem of "scale" arises in full force; and this makes the problems of computational language understanding in no way less difficult.

5.5. Networks for Parsing and Preferences

One crucial component in a parsing system that operates over real world text is the module for making informed preference evaluations. Certain areas of the immense search space can be pruned or ignored on the basis of syntactic or other constraints as, for example, when the choices

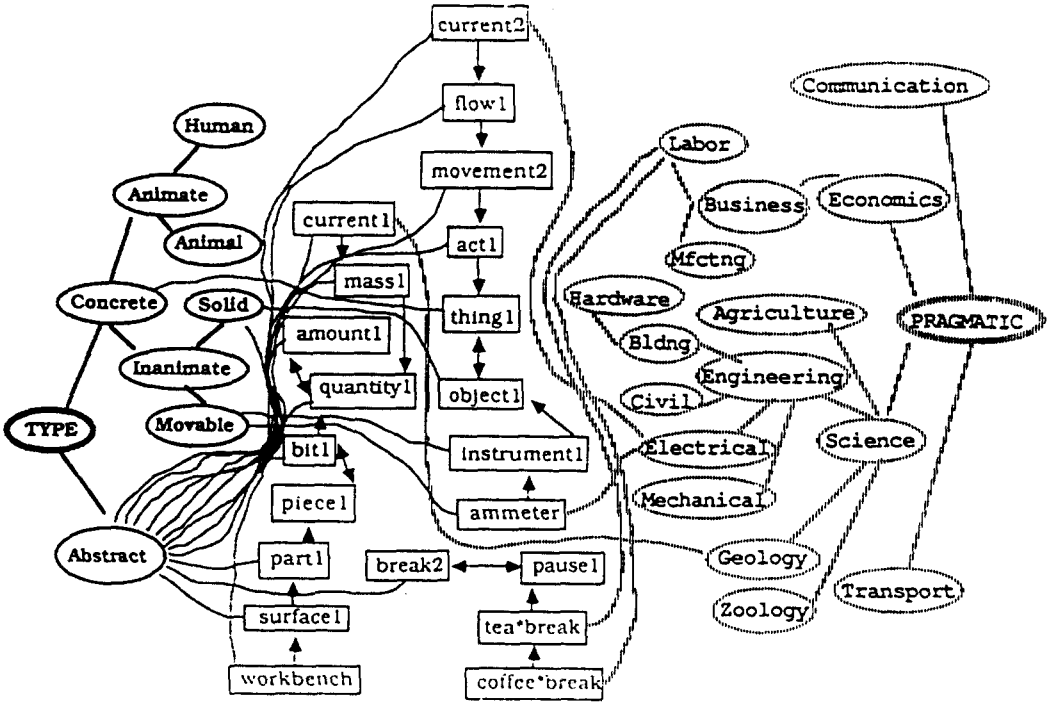

Figure 2. Some network structure connected to sentences S1 and S2.

for "measure" in sentence S1 are reduced from 26 to 11 by simply observing that only "measure" can be the verb of the sentence. Of course, this is something of an "after the fact" constraint, since a purely left-to-right parser cannot know until later in the sentence that "measure" must be a verb; but either sufficient look-ahead or proper back-tracking will deterministically resolve the issue in this case.

However, other crucial decisions amount to ranking alternatives at a given point and searching them in "best-first" order, since even the eleven-way choice at "measure" can lead to considerable processing overhead. This preference evaluation is achieved in PREMO with a complex function that takes into account the grammatical, semantic, pragmatic, and inferential information captured in the dictionary network structures described above, and as shown in Figure 2.

5.6. Grammatical Predictions

The lexical entries in the network are classified as to grammatical type, according to the LDOCE grammar coding system described above. Further, there are grammatical predictions encoded as well. For example, the 4th sense of the noun "measure" requires that it be followed by "of"; as in "a measure of success." The existence of a "requirement" link in the representation for this lexical item allows it to be ranked very poorly by the preference evaluation function once it becomes clear that "of" does not follow "measure" in sentence S1. In the general case, where there are other items in the sentence that could be parsed as verbs, thus making the noun senses of "measure" viable candidates, the failure or success of these grammatical predictions are important for parsing.

5.7. Semantic Matching

The verbs in LDOCE encode TYPE preferences for their subjects and objects, and the nouns and pronouns express their TYPE in the same way. For example, the first two verbal senses of "measure" prefer a *human* subject, while the third sense prefers a subject that is *solid*. The 2 pronoun senses of "he" express themselves as either "H" (for *human*, identically with the TYPE code in "measure") or "K" (for *human-male+animal-male*). Therefore, the preference evaluation function will rank the first two senses of "measure" very highly over the third one. This is because the semantic distance across the TYPE network is much less, and "nearness" in the TYPE coding system is one way to measure and compare semantic preferences.

5.8 Pragmatic Coherence

A significant fraction of the lexical items in LDOCE are tagged with a PRAGMATIC code, and it is a feature of language in general that elements in a text will tend to be coherent with respect to the subject area of the text. That is, if the topic of a text is, say, electricity, then we would expect the word "current" in that text would often be used in the electrical rather than the river sense. This is not a hard and fast rule, and exceptions are easy to construct, but it is an intuitive notion about language that will hold in the main.

In sentence S1, for example, the word "ammeter" in LDOCE is defined in a single sense, and that sense is marked with the PRAGMATIC code for **engineering/electrical**. The word "current" is defined in LDOCE with 7 senses, of which only the 2nd and 3rd noun senses are marked with the **engineering/electrical** code. The preference evaluation function takes this into account during parsing, and this is the crucial word-sense distinction pointed out in the natural language understanding section, above. A good choice at this point permits a question-answering system to make a good response to the question, "What is current?".

5.9 Inferential Reasoning

Structural ambiguities arise in the attachment of prepositional phrases. In sentence S1, the phrase "on his workbench" can either be the TIME of the "measuring," as in "on his coffee break", the LOCATION of the "measuring," or the LOCATION of the "ammeter." The TIME interpretation can be easily discounted, because the TYPE associated with "workbench" is *solid*, and this will not fill a TIME role. However, the choice between the two LOCATION cases is more problematic.

The object of the "measuring" in sentence S1 is the **electrical** sense of "current," whose genus is "flow." The correct sense of "flow" in LDOCE is the 2nd one, which is marked as an *abstract* sense defined as "a smooth steady movement." These associations are made by the Genus Disambiguator program, described in Section 3.1 on implicit dictionary structure, above.

Preference evaluation would expect to find the LOCATION of the "measuring" of "current" to ideally be on an abstract electrical place, such as a "circuit," and this is not possible to arrive at from the representation of "workbench." Meanwhile, preference evaluation has no trouble associating "workbench" which is a *solid* whose genus is "a surface" as the LOCATION of "ammeter," which is a *movable-solid* whose genus is an "instrument," which is also a *movable-solid*.

This association is reinforced by matching the PRAGMATIC codes of the genus of "ammeter," which is "instrument1," and the PRAGMATIC codes of "workbench," which are both **hardware/tools**. Therefore, the preference evaluation function will choose to rank the "workbench" as the LOCATION of the "ammeter" over the "workbench" as the LOCATION of the "measuring," since this is most inferentially plausible in terms of the connectivity of the network of primitive elements and word senses.

6. CONCLUSION

Parsing text with the aid of machine-readable lexicons poses all the usual problems of NLP, plus additional problems of scale. Machine-readable dictionary research helps to mitigate the vocabulary problem, but itself introduces increased lexical ambiguity. On the other hand, when well defined networks of primitive elements and word senses are derived from machine-readable dictionaries, these provide the structured knowledge that is necessary for understanding natural language in the face of lexical and structural ambiguity.

REFERENCES

1. *Longman Dictionary of Contemporary English (LDOCE)*, (Edited by P. Procter *et al.*), Longman Group Ltd., Harlow, Essex, UK, (1978).
2. B.M. Slator, Using context for sense preference, *Proceedings of the First International Lexical Acquisition Workshop (IJCAI-89)*, August 21, Detroit, MI, Chapter 13, pp. 1-8, AAAI Press, (1989).
3. R.H. Fowler and B.M. Slator, Information retrieval and natural language analysis, *Proceedings of the 4th Annual Rocky Mountain Conference on Artificial Intelligence (RMCAI-89)*, June 8-9, Denver, CO, pp. 129-136, (1989).

4. R.A. Amsler, A taxonomy for English nouns and verbs, *Proceedings of the 19th Annual Meeting of the Association for Computational Linguistics*, Stanford, pp. 133–138, (1981).
5. R.A. Amsler, The structure of the Merriam-Webster Pocket Dictionary, Technical Report, (TR-164), Ph.D. Thesis, University of Texas at Austin.
6. R.A. Amsler and J.S. White, Development of a computational methodology for deriving natural language semantic structures via analysis of machine-readable dictionaries, NSF Technical Report, (MCS77-01315).
7. *The New Merriam-Webster Pocket Dictionary*, Pocket Books, New York, (1964).
8. J. Nakamura and M. Nagao, Extraction of semantic information from an ordinary English dictionary and its evaluation, *Proceedings of COLING-88*, Budapest, Hungary, pp. 459–464, (1988).
9. M.S. Chodorow, R.J. Byrd and G.E. Heidorn, Extracting semantic hierarchies from a large on-line dictionary, *Proceedings of the 23rd Annual Meeting of the ACL*, Chicago, IL, pp. 299–304, (1985).
10. *Webster's Seventh New Collegiate Dictionary*, C. & C. Merriam Company, Springfield, MA, (1967).
11. L. Guthrie, B.M. Slator, Y. Wilks and R. Bruce, Is there content in empty heads?, *Proceedings of the 13th International Conference on Computational Linguistics (COLING-90)*, Aug. 20–25, Helsinki, Finland, (1990).
12. B.M. Slator, Constructing contextually organized lexical semantic knowledge-bases, *Proceedings of the Third Annual Rocky Mountain Conference on Artificial Intelligence (RMCAI-88)*, June 13–15, Denver, CO, pp. 142–148, (1988).
13. B.M. Slator, Extracting lexical knowledge from dictionary text, *Knowledge Acquisition: An International Journal (KAAIJ)* 1 (1), 89–112, Academic Press, London (March 1989a).
14. B.M. Slator and Y. Wilks, Towards semantic structures from dictionary entries, *Proceedings of the Second Annual Rocky Mountain Conference on Artificial Intelligence (RMCAI-87)*, June 17–19, Boulder, CO, pp. 85–96, (1987).
15. Y. Wilks, D. Fass, C. Guo, J.E. McDonald, T. Plate and B.M. Slator, Machine tractable dictionaries as tools and resources for natural language processing, *Proceedings of the 12th International Conference on Computational Linguistics (COLING-88)*, Aug. 22–27, Budapest, Hungary, pp. 750–755, (1988).
16. Y.A. Wilks, D.C. Fass, C. Guo, J.E. McDonald, T. Plate and B.M. Slator, A tractable machine dictionary as a resource for computational semantics, In *Computational Lexicography for Natural Language Processing*, (Edited by B. Boguraev and T. Briscoe), pp. 193–228, Harlow/ Longman, Essex UK,, (1989).
17. Y.A. Wilks, D.C. Fass, C. Guo, J.E. McDonald, T. Plate and B.M. Slator, Providing machine tractable dictionary tools, *Machine Translation*, (Edited by S. Nirenburg) 5 (2), 99–151 (1990).
18. N. Varile, Charts: A data structure for parsing, In *Parsing Natural Language*, (Edited by M. King), pp. 73–87, Academic Press, London, (1983).
19. C. Guo, Constructing a machine tractable dictionary from Longman Dictionary of Contemporary English, Computing Research Laboratory Memoranda in Computer and Cognitive Science (MCCS-89-156), Doctoral Dissertation, New Mexico State University (1989).
20. J. Markowitz, T. Ahlswede and M. Evens, Semantically significant patterns in dictionary definition, *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics*, pp. 112–119, New York, (1986).
21. Y.A. Wilks, *Grammar, Meaning, and the Machine Analysis of Language*, Routledge and Kegan Paul, London, (1972).
22. Y.A. Wilks, An intelligent analyzer and understander of English, *Communications of the ACM* 18 (5), 264–274 (1975). Reprinted in *Readings in Natural Language Processing*, (B.J. Grosz, K. Sparck-Jones and B.L. Webber, Eds.), pp. 193–203, Morgan Kaufmann, Los Altos, CA, (1986).
23. Y.A. Wilks, A preferential pattern-seeking semantics for natural language inference, *Artificial Intelligence* 6 (1), 53–74 (1975).
24. Y.A. Wilks, Making preferences more active, *Artificial Intelligence* 11 (3), 75–97 (1978).
25. B.M. Slator, PREMO: The PREference machine organization, *Proceedings of the Third Annual Rocky Mountain Conference on Artificial Intelligence (RMCAI-88)*, June 13–15, Denver, CO, pp. 258–265, (1988).
26. B.M. Slator and Y. Wilks, PREMO: Parsing by conspicuous lexical consumption, *Proceedings of the International Workshop on Parsing Technologies*, August 28–31, Pittsburgh, PA, Carnegie Mellon University, pp. 401–413, (1989).