Lexicographic Support for Knowledge-Based Machine Translation

SERGEI NIRENBURG
Carnegie Mellon University, Pittsburgh, USA

1. Lexicographic needs of KBMT systems

The general control strategy of a typical knowledge-based machine translation system is as follows. At the first stage the propositional-semantic and the discourse-pragmatic meanings of a source language (SL) text are extracted and represented as *texts* in a formal language-independent notation, traditionally called interlingua (IL). The IL is the knowledge representation language in which the result of SL text analysis is written. The result of this analysis stage for a given SL text is an IL text (ILT). After an ILT is produced it is then augmented, through the operation of the inference-making component of a KBMT system to make the amount of information it carries adequate for the requirements of the generation process. Finally, the augmented ILT is sent to the generator which produces a text in the target language (TL). At this level of abstraction, the process is in accordance with the knowledge-based machine translation paradigm introduced in (Carbonell *et al.* 1981).

Conceptually, the three processes mentioned above are supported by three dynamic knowledge sources: the analyzer, the augmentor, and the generator. In practice, however, the responsibilities of the augmentor can be (at least partially) subsumed by the analyzer itself. To operate successfully, the above processing modules rely on the knowledge stored in the static knowledge sources (SKS) of a KBMT system. The latter can be classified as follows:

1. SKSs that store natural language-dependent information; these include
   (*a*) syntactic grammars (for each SL and TL)[1]
   (*b*) a bilingual (SL − IL) analysis lexicon (AL) that associates units of a natural language with units of the interlingua; needed for each SL
   (*c*) a bilingual (IL − TL) generation lexicon (GL) that associates units of the interlingua with those in a natural (TL) language; needed for each TL
2. SKSs that store language-independent information; these include
   (*a*) the world concept lexicon (CL) that stores the knowledge about types of conceptual entities present in the subworld corresponding to the subject domain of translation and their properties; thus, for example a description of concepts such as 'material' or 'computer' and properties such as 'specific gravity' or 'color' will belong here
   (*b*) the long-term fact repository of the system that stores knowledge about particular (remembered) tokens of the entity types; thus, for example, the fact that the name of the laser printer at the CMU Center for Machine Translation is 'Tara' and that it is located in Room 109B will be stored here (while the knowledge about the concept 'laser printer' will be stored in the CL!)
   (*c*) the IL text used to describe the knowledge about the events mentioned in the current SL text, as well as about the nonpropositional information obtained from the SL text as a result of the operation of the analyzer.[2]

Not all of the above knowledge sources will be equally important for all types of subject domains. In particular, the episodic memory is the least developed of the knowledge representation areas, due to the fact that the domain of computer manuals we are exploring does not seem to require more long-term knowledge about tokens than that stored in the intermediate memory. Other, more 'case-based' domains, such as those of legal texts or medical histories and doctor-patient communication, require a temporally organized episodic memory.

The approach discussed in this paper has been tested in a large knowledge-based machine translation system, KBMT-89, developed at the Center for Machine Translation of Carnegie Mellon University. Goodman and Nirenburg (1989), is a detailed report about this system.

2. The format of IL text

IL texts are the central, pivotal knowledge source in the process of KBMT. Indeed, they interface between the processes of analysis and generation. The quality and detail of ILTs are crucial to the success of the translation process. Unlike a natural language text, an ILT is not linear. It is a multiply interconnected set of knowledge structures that correspond to

- instances of domain (translation subworld) concept types (*events* and their actants, or *roles*) mentioned in the SL texts, and
- instances of text structure component types that capture the *manner* in which the above propositional knowledge was expressed in SL (these structure component types are IL Sentences and IL Clauses, and they are linked through *discourse cohesion markers*).

An IL sentence is represented as a frame with slots for each of any number of IL clauses (that are represented as frames themselves) as well as for speech act and focus information. The IL clause is the place where events (act instances) are put into their modal, discourse, and spatio-temporal context. Events and roles that appear in ILTs

Correspondence: Sergei Nirenburg, Center for Machine Translation, Carnegie Mellon University, Pittsburgh, PA 15213, USA.

Literary and Linguistic Computing, Vol. 4, No. 3, 1989

are produced instantiating tokens of the appropriate concept types in the concept lexicon and augmenting them with various property values identified during SL text analysis (see an example below). It follows that the slots whose values express a component of contextual propositional meaning (e.g. negation) or any type of nonpropositional, including discourse, meaning appear only in IL text frames for event and object tokens, and *not in* the concept lexicon.

Every token stands in the *is-token-of* relationship to its corresponding type. The frame for a type and the frame for a corresponding token are, however, not necessarily identical either in structure or semantics, even though they share many slot names. Still, there are regular correspondences between units of CL and ILT. The property values in concept tokens are typically elements or subsets of data types listed as fillers for the corresponding slots in CL frames. Thus, for instance, the *color* property slot in the CL frame for the concept of *rose* can be occupied by a list (*white yellow blue red purple* ...). At the same time, the ILT frame representing a particular rose, say, *rose11*, will have the value *red* as the contents of its *color* slot. To appreciate the complexity of the task of adequately describing the world based on knowledge about 'typicalities', consider the case when the value of a slot for a token will be from outside the range suggested for the corresponding type. Thus, the *color* slot for the concept type of *snow*, will contain the one-element list (*white*). But a particular instance of (a quantity of) snow can be actually *grey*. It is arguable, however, that from the standpoint of an inference-making mechanism it is preferable to accept such default overridings and not to 'dilute' the prototypical value of color with the seldom used *grey*.

The nonpropositional knowledge derived from the NL input is also represented in ILT. IL sentences, clauses, and events carry this information as appropriate. The overt representation of ILT units, not just a sequence of events, is one of the principal technical innovations of our approach. Note that for representing nonpropositional relations (such as discourse or focus) we use the same knowledge structures that are more traditionally used for representing the propositional content of NL input. Definitions of entities at the higher levels of ILT syntax follow. The definitions are edited to highlight the relevant features.

```
text ::= sentence | (discourse-structure-type text text+)

sentence ::= ((id string)
              ('clauses' clause+)
              ('subworld' subworld)
              ('modality' modality)
              ('focus' focus)
              ('speech-act' speech-act))
```

Thematic information about the sentence includes the values for the *given* and the *new* (or focus) slots in the sentence frame. Both values can be pointers to a concept, a property of a concept, or an entire clause. The value of the modality slot for the IL sentence is chosen from the set of modalities.

```
clause ::= ((id string)
            ('discourse-structure' discourse-structure)
            ('focus' focus)
            ('modality' modality)
            ('time' time)
            ('space' space)
            ('events' event)
            ('quantifier' quantifier2)
            ('subworld' subworld))
```

Discourse structure markers can connect a clause not only with another clause but also with an object or an event, as well as with a sentence, a paragraph or even a whole text.

```
modality ::= ('modality' modality-sort

modality-sor :: = real | desirable | undesirable | conditional |
                  possible | impossible | necessary

focus ::= ('given'
                  ('object' obj) |
                  ('event' event) |
                  ('clause' clause) |
                  ('quantifier' event-quantifier | quantifier))
          ('new'
                  ('object' obj) |
                  ('event' event) |
                  ('clause' clause) |
                  ('quantifier' event-quantifier | quantifier))

discourse-structure ::= (discourse-structure-type
                          (clause1  clause-n | sentence | text) |
                          (clause-n | sentence | text  clause1)* )

discourse-structure-type ::= none | temp | equiv | purpose | -purpose ||
                             condi | +elmit | -elmit | choice
```

For a more detailed description of the discourse cohesion markers we use see Tucker *et al.* (1986).

```
speech-act ::= ('speech-act'
                                ('type' speech-action)
                                ('direct?' yes | no)
                                ('speaker' object)
                                ('hearer' object+)
                                ('time' time)
                                ('space' space))
```

The time and space of the speech act can be quite different from that of the proposition which is the information transferred through this speech act.

3. The concept lexicon

CL is the primary source of building blocks for which ILTs are constructed. It plays the part of the system's model of its world, the ontological descriptions of all the objects, acts, and relations of which the system is aware. Therefore the creation of a sizable CL must precede work on building ALs and GLs, since the CLs are the basis of the construction of ILT.

The concept lexicon is organized as a set of entries (concept type nodes), represented as frames, and a set of links, represented as slots in these frames. Slot fillers are pointers to concept nodes.[3]

The structural backbone of CL is made up of the tangled *isa* hierarchy with property inheritance, the transitive partonomic *part-of* hierarchy, and the empirical, semantic field-like classification of all concepts into subject domains or subworlds.[4] The taxonomy and the partonomy are examples the paradigmatic relationships among concept *types*. Case-frame, precondition/effect and ownership and some other links describe syntagmatic relationships that tokens of a particular type have with tokens of other types in an IL text.

The semantics of syntagmatic relationships in CL is as follows: the corresponding property values represent either defaults or acceptable value ranges (cf. the above examples of the *color* values for *snow* and *rose*, respectively); they are used to guide the preference-based (Wilks, 1975) treatment of the disambiguation process and for validity checking. Concept *tokens*, being components of ILT, not CL, have their slots occupied by *actual values of properties*; if information about a property is not forthcoming in the input, then the default value (if any) is inherited from the corresponding type representations. That is, if nothing is said in the text about the color of the particular quantity of snow, we will assume

that it is white. Some examples of IL lexicon frames are given below.

```
all ::= {'all'
         ('subworld' subworld*)}
```

This is the root of the isa hierarchy. Note that we believe that every node in the lexicon represents a concept that belongs to one or more subworlds.

```
process ::= {'process'
             ('isa' all)
             ('partof' object*)}

action ::= {'action'
            ('isa' process)
            ['has-as-part' process-sequence]
            ['part-of' process*}
            ['agent' creature]
            ['object' object]
            ['instrument' object]
            ['source' object]
            ['destination' object]
            ['precondition' state*]
            ['effects' state*]}
```

The 'has-as-part' slot of an action frame contains either the constant *primitive*, if the process is not further analyzable in IL lexicon, or an expression in a special temporal/causal process description language.[5]

```
physical-action ::=
        {'physical-action'
         ('isa' action)
         ['object' object]}

mental-action ::=
        {'mental-action'
         ('isa' action)
         ['agent' creature]
         ['object' object | process]}
```

Mental actions further classify into reaction actions (cf. the English 'please' or 'like'), cognition actions ('deduce'), and perception actions ('see'). Objects of mental actions can be either objects, as in (3), or processes, as in (4).

(3) I know John

(4) I know that John has traveled to Tibet

```
speech-action ::=
        {'speech-action'
         ('isa' action)
         ['agent' person]
         ['patient' person* | organization*)
         ['object' event | object]
         ['source' 'agent']
         ['destination' 'patient']}
```

The 'agent' slot filler for speech actions has the semantics of the speaker. The 'patient' is the hearer.

```
state ::= {'state'
                    ('isa' process)
                    ['part-of' state*]}
```

The actant in states, which is the patient rather than the actor, is inherited from the process frame.

```
object ::=
        {'object
         ('isa' all)
         ['part-of' object*)
         ['has-as-part' object*]
         ['belongs-to' creature | organization])
```

4. The analysis lexicon

To represent the analysis lexicon one first has to suggest a representation language for the IL texts, because some of the AL entries will be formulated as instructions to record certain information in the IL text directly, without the mediation of the concept lexicon.

4.1. *Analysis lexicon entries—Type I*

Type I AL entries connect units of SL with concepts in IL:

DATA data

The CL definition of data is as follows:

```
(data
 (isa information)
 (subworld computerworld mathematical world)
 (object-of computer-mental-action)
 (instrument-of mental-action)
 (belongs-to user)
 (consists-of file record byte)
 (part-of database))
```

4.2. *Analysis lexicon entries—Type II*

Type II AL entries are commands to insert a value into one of the property slots of existing concept-token frames (the entry is described informally).

PERMANENTLY no corresponding CL concept; insert the value 'always' in the time property slot of the event which this word modifies.

4.3. *Analysis lexicon entries—Type III*

Type III AL entries contain test or control knowledge for the analyzer decisions, as illustrated by one of the AL entries for THE:

THE no corresponding CL concept or property; expectation: an NP follows; anaphor resolution heuristic: this NP is coreferential with a concept token from among the concept tokens in the analysis workspace for the current text.

5. The generation lexicon

5.1. *Requirements for lexical realization*

Languages 'view the world' in different ways.

Table 5.1 Languages view the world differently.

IL	Hungarian	French	Malayan
Elder brother	batya	frere	sudara
Younger brother	ocs	frere	sudara
Elder sister	nene	soeur	sudara
Younger sister	hug	soeur	sudara

Table 5.2 In-law terms in Russian.

English	Russian
wife's father	test'
wife's mother	tyoshcha
husband's father	svyokor
husband's mother	svekrov'
wife's brother	shurin
wife's sister	svojachenitsa
husband's brother	dever'
husband's sister	zolovka
brother's wife	nevestka
sister's husband	zyat'
son's wife	snoxa
daughter's husband	zyat'
daughter's husband's father	svat
daughter's husband's mother	svatja
son's wife's father	svat
son's wife's mother	svatja

The above example (from Hjelmslev) means that when Malayan sudara is analyzed, the information on relative age and sex of the sibling has to be sought through additional inference-making—provided IL stipulates the grain size that requires this level of detail. Generating

Malayan from an internal representation will be relatively simpler, since, if the information about sex and relative age is not in focus, it can be omitted, thus simplifying the lexical selection. With Hungarian, it will be the generation side (and, therefore, the generation lexicon that will require additional information for lexical selection), while analysis will be relatively straightforward.

We assume that the grain size for the descriptions in the concept lexicon is determined empirically based on language-independent considerations of practical necessity. This means that with some languages the analysis side will be more involved, with other languages, the generation side.

5.2. *The structure of a GL entry*

The structure of an entry in the generation lexicon is described below in the BNF. The BNF is incomplete, wherever obvious.

```
<GL-entry>          ::= ( <meaning-pattern> <TL-pattern>* )
<meaning-pattern>   ::= [ [! SEL-slot ] <SEL-slot slot-value> ]* )
<TL-pattern>        ::= ( <TL-lexeme> <lex-info> <collocation> )
<TL-lexeme>         ::= ( <language> <L-lexical-unit> )
<language>          ::= english | spanish | russian | japanese | ...
<lex-info>          ::= ( lexical-class <lexical-class> )
                        <syntactic-info> {morph <inflection-type> }
<lexical-class>     ::= noun | adjective | verb | ...
<syntactic-info>    ::= {the usual contents of syntactic dictionary}
<inflection-type>   ::= {an indication of irregularities in forming
                        word forms, e.g., "goose" - pl. "geese," etc.}
<collocation>       ::= ( [! <dimension> <dimension-value>* )]* )
<dimension>         ::= {a set based on the appropriate representation
                         language/slot names for the concept in question}
<dimension-value>   ::= {a TL lexical unit (word or expression) that can
                        be ordinarily found in text around the TL
                        lexicalizant in <TL-lexeme> above; can be
                        recursive, see the second example below}
```

What the above means is that the GL entry consists of

(1) a frame which represents the meaning of a TL lexical unit and is used as index for matching with input (IL Text) frames during the lexical selection stage of the generation process;

(2) a set of 'right-hand sides' (no more than one RHS per TL, but some TLs may not have a RHS for a GL entry); a RHS consists of a set of lexical units (TL synonyms) that correspond to the meaning of the entry header, their grammatical classification, and their collocation characteristics.

Natural languages influence lexical selection in ways that are sometimes 'illogical', that is not readily explainable through semantic distinctions. Why do we use, in English, *shed* with *tears* or *leaves* but do not usually say *shed water out of a bucket* or *they drop tears every time when ...*?

As another example, consider the conceptual operator of *a large quantity of*, a (relative) value for measuring quantities (of materials, forces, qualities, properties, etc.). It is realized in English in accordance with collocational properties of the lexical units involved.

voltage	great
difficulty	wide
wind	enormous
selection	large
amount	big
expanse	strong

Consider one more example. The first of the two lists below shows the choice of English realizations for the concept of, roughly, *growing/getting smaller*. The second list contains examples of quantities that can get smaller. Suppose that the sentence type selected by a natural language generator is such that the members of the second list will become subjects of corresponding clauses (e.g. *temperature decreased*). Not all of the words in the first list can be used with those in the second list. It seems difficult (if at all possible) to distinguish the meanings of the elements of the first list so that they can be 'automatically' selected provided the meaning underlying a particular element of the second list is present in ILT.

decrease	temperature
drop	crops
diminution	dollar (value of)
shrinking	profit
decline	power
reduction	cloth (area of)
deflation	crime

Collocation properties of a lexical unit effectively build a lexical-semantic field around it; it is possible that a more efficient way of delineating these lexical fields will be found than listing the lexical units from one field multiply in the entries devoted to each and every one of them. The lexical field ideas stem from seminal work by Mel'cuk and his various co-authors. See, for instance, Mel'cuk (1982).

The above design has been implemented in the Diogenes natural language generation system. See Nirenburg *et al.* (1988*a*); and Nirenburg *et al.* (1988*b*) for a detailed description.

Figure 5.2 summarizes our approach to lexicons in KBMT and their influence on the creation and manipulation of the interlingua text. It shows the fragment of the AL necessary to process the sentence: 'Some data, however, may be lost.'

```
(((is-token-of person)  ;<meaning-pattern>
  (sex male)
  (age (range 2 15)))
 ((english "boy")
  (lexical-class noun (noun-type class)(morph regular))
  (collocation (place ("school" "playground"))
               (agent-of "do homework" "play ball")
               (instruments "pen" "notebook" "ball")
               (pat-complement-pos "girl")
               (pat-complement-age "baby" "man"))))


(((is-token-of person)
  (age (range 2 15))
 ((english "child"
  (lexical-class noun (noun-type class)(morph (plural "children")))
  (collocation (place ("home" "playground"))
               (agent-of "play")
               (instruments "toy")
               (object-of ((relat" (agent "parent" "mother" "father"))
                           ("educate" (agent "teacher")))
               (pat-complement-age "adult" ))))
```

Fig. 5.1 Sample entries in the generation lexicon

6. Getting there: lexicographic knowledge acquisition

One of the main lexicon-related tasks in KBMT is the acquisition of the lexicons. We have come up with the idea of a Lexicon Management System (LMS), an interactive aid for lexicon acquisition that will help to acquire the concept lexicon for the subject domain(s) of translation as well as an analysis lexicon for each SL and a generation lexicon for each TL (see Nirenburg and Raskin (1987)), for a more detailed description of the

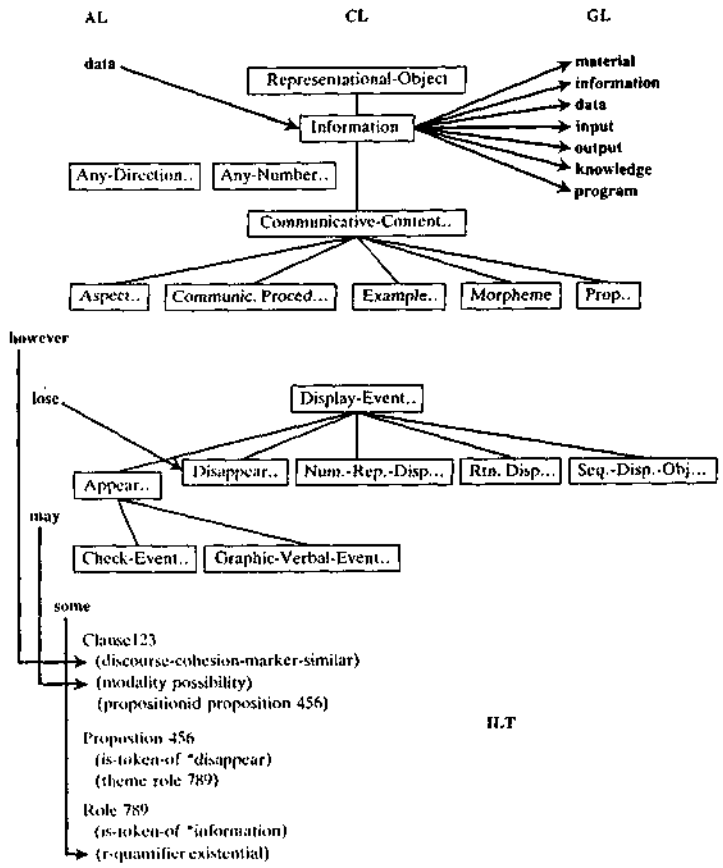

Fig. 5.2. The interaction among lexicons and ILT. Note that some subject language lexical units are connected to their interlingua meanings directly, bypassing the concept lexicon. The figure also illustrates the lack of symmetry in the treatment of lexical semantics in analysis and generation; the main problem in analysis is polysemy, while in generation it is synonymy

LMS functionality. The LMS has to support a variety of functionalities:

- a structured screen editor customized for the knowledge acquisition and maintenance requirements of a particular application
- a graphics-based browser
- a natural-language interface to support information retrieval for knowledge acquisition and maintenance
- a means of interaction among the above functionalities (modes of operation).

Since the LMS is envisaged as a 'smart' tool, additional functionalities must be provided:

- a variety of consistency and validity checks on the user-supplied information
- a flexible and extensive on-line help and 'suggestions' facility
- a means of effective communication among the members of the knowledge acquisition team.

Examples of 'low-level' functionalities supported by the LMS include:

- Displaying and manipulating texts in multiple scripts (e.g. Latin, Cyrillic, Hebrew, Arabic, Kanji) on a single screen.
- Handling diacritical symbols, right-to-left and top-down writing styles, etc.
- Automatic conversion of a text in a foreign script into a canonical internal ASCII representation and vice versa.
- Advanced and user-friendly editing capabilities, including searching, scanning, moving big chunks of text around, etc.
- Advanced and user-friendly formatting capabilities,

including footnotes, 'invisible' comments, footers and headers, font sizes and styles, section numbering, indexing, bibliography, and table of contents preparation, etc.

- Spelling checking and (simple) text critiquing.

Our initial implementation of a LMS is the Ontos knowledge acquisition and maintenance system (Nirenburg *et al.*, 1988*b*), which supports a large subset of the above functionalities. Ontos has been extensively used in the development of lexicons for KBMT-89.

Acknowledgements

I would like to thank Jaime Carbonell, Roland Hausser, Ira Monarch, Eric Nyberg, Victor Raskin, Rich Thomason, and Michael Witbrock for their comments and suggestions.

Notes

1. It is debatable whether the identical grammar should be used for analysis and for generation. Clearly the processes are different, but if both use the same grammar provisions must be made for comprehending a larger range of sentences than those generated, which normally adhere to stricter principles of grammatical correctness—or at least preferences for grammatical normalcy.
2. To use psychological terminology, all of 1, 2(*a*) and 2(*b*) belong to the 'long-term memory' of the system, while 2(*c*) belongs to the 'short-term' or 'intermediate' memory. Within the long-term memory 2(*b*) belongs to the 'episodic' memory, while all the rest, to the 'semantic' one (cf. Tulving, 1985). The latter stores knowledge about *types* of concepts in the world and types of language constructs, while the former is a repository of facts about *tokens* of particular types.
3. In a practical simplification, we allow some properties to take values from specially defined value sets; this device is used temporarily, until the corresponding part of the concept network gets fully developed.
4. Being primarily concerned with descriptive adequacy in a subject domain, we do not enforce an a priori limit on the number of (either primitive and derived) concepts and properties we use.
5. For a detailed discussion see Nirenburg *et al.* (1986).

References

Carbonell, J., Cullingford, R., and Gershman, A. (1981). 'Steps Towards Knowledge-based Machine Translation', *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 4, 376–92.

Goodman, K. W. and Nirenburg, S. (1989). KBMT-89. Project Report. Center for Machine Translation, CMU. February.

Mel'cuk, I. (1982). 'Lexical Functions in Lexicographic Description', In *Proceedings of the 8th Annual Meeting of Berkeley Linguistic Society*.

Nirenburg, S. and Raskin, V. (1987). 'The Subworld Concept Lexicon and the Lexicon Management System', *Computational Linguistics*, 276–89.

—— —— and Tucker, A. (1986). 'On Knowledge-Based Machine Translation', *Proceedings of International Conference on Computational Linguistics*, 627–32. COLING-86, Bonn, Germany, August 1986.

—— Nyberg, E., McCardell, R., Huffmann, S., and Kenschaft, E. (1988*a*). *DIOGENES-88* TR-88-107, Center for Machine Translation, Carnegie Mellon University.

—— Monarch, I., Kaufmann, T. and Nirenburg, I. (1988*b*). 'Acquisition and Maintenance of Very Large Knowledge Bases', TR-88-108, Center for Machine Translation, Carnegie-Mellon University.

Tucker, A., Nirenburg, S. and Raskin, V. (1986). 'Discourse, Cohesion and Semantics of Expository Text', *Proceedings of COLING-86*, 181–3.

Tulving, E. (1985). 'How Many Memory Systems Are There? *American Psychologist*, 40, 385–98.

Wilks, Yorick (1975). 'A Preferential Pattern-Seeking Semantics for Natural Language', *Artificial Intelligence*, 6, 53–74.