Towards a Peircean model of language

Guy Debrock and Janos Sarbo

University of Nijmegen, Toernooiveld 1
6525 ED Nijmegen, The Netherlands

Abstract

We argue that traditional approaches to natural language suffer from the 'fallacy of misplaced concreteness'. Because 'language' is a noun, and nouns usually refer to 'things', it is often assumed that language is some 'thing' with a certain immutable structure and properties. This problem of language modelling is also witnessed by the limited success of phrase structure-based parsers in natural language processing. One reason for this lies in the rigidity of hierarchical structure on the one hand, as opposed to the high flexibility of language use on the other.

It will be argued that language is in the first place a process, and that this assumption puts the task of an analysis of language in a different perspective. A model supporting this view is Natural Language Concept Analysis (NLCA). In NLCA, hierarchical structure is found as the result of the interaction of different, inherent combinatorial properties of linguistic units. The purpose of the paper is to show that NLCA is consistent with C.S. Peirce's pragmatic, evolutionary and semeiotic approach, and that such approach supports and clarifies NLCA.

1 Introduction

In this paper, it will be argued that an adequate analysis of natural language requires a change of perspective regarding the nature of language, and that this change of perspective in turn requires a change of ontological perspective. A paradigmatic new perspective has also been encouraged by the limited success of traditional approaches in natural language processing (e.g. [Hae91], [PS94], [Lam88]). In such approaches some form of hierarchical structure (e.g. phrase structure) plays a central role. One

reason for the relative lack of success with rule-based parsers is due to the discrepancy between the rigidity of hierarchical structure and the high flexibility of language use. The growing realisation of this problem has inspired a search for alternative methods, such as statistical-based and lexicon-driven parsing.

Natural Language Concept Analysis (NLCA) introduced in [KS98] takes a different approach in which hierarchical structure is found to be the result of the interaction of different, inherent combinatorial properties of linguistic units.

The first part of this paper is an attempt to offer a Peircean explanation of the NLCA-approach. More specifically, we would like to show how Peirce's categoreal and semeiotic approach may provide a theoretical framework that is consistent with and supportive of NLCA. After a brief exposition of the thesis that existing approaches to language tacitly suppose a substantialist ontology, we briefly state the conditions that are necessary to an adequate approach of language. This is followed by an attempt to show that Peirce's categoreal scheme meets those requirements, provided it is complemented by a Peircean theory of signs. In the last section of this first part, an attempt is made at providing a Peircean interpretation of syntax which is consistent with the basic insights of NLCA.

The focus of the second part is directed to NLCA itself. First, relation schemes implementing the Peircean semeiotic view of language are described. This is followed by a section devoted to the algorithmic and representational aspects of NLCA; the section is completed by an example. We end the paper by discussing the question how the results of an analysis in NLCA can be represented as conceptual structures.

2 The fallacy of misplaced concreteness

Linguistics has its origin in Greek and Medieval tradition. The basic presupposition of that tradition is that everything is to be explained in terms of things (substances) and their properties: not only qualities and quantities, but also action and passion are real by virtue of their being attributed to substantial entities. This position which was first explicitly stated by Aristotle was derived from an ingenious analysis of the Greek language, in which every well-formed sentence seems to consist of a reference to something of which something else is said. Indeed, Aristotle ([Ari84a]) defined a substance as something of which many things can be said, but which itself can never be said of another thing. In other words,

substance is whatever can never function as a predicate.

It soon appeared, however, that a number of phenomena could not be described either in terms of things nor in terms of properties. But because they were signified by (substantial) nouns, those phenomena were approached as if they were things, as quasi-things. Such was the case of, for instance, nature, cause, life, truth, and language. Thus, from the fact that language is a noun, it is usually inferred that it must be seen as a quasi-thing with certain properties. A.N. Whitehead ([Whi85]) used the expression "the fallacy of misplaced concreteness" to designate the – usually unnoticed – mistake of treating what is not a concrete entity as if it were a concrete entity. In as much as language was studied as a quasi-thing while it is not, it may be said that the traditional approach of language suffers from the fallacy of misplaced concreteness. In the case of language, the fallacy consists in attributing to language an immutable and universally valid structure.

Another aspect of the traditional approach, is that it is ambiguous regarding the natural status of language. On the one hand, it is generally considered to be a natural phenomenon, on the other hand, it is considered as something intrinsically human. Indeed, Aristotle ([Ari84b]) would define the human being as ζῶον λόγον ἔξον, an animal with the particular characteristic of having the ability of collecting (literally, *reading*) elements which by themselves have no meaning into a meaningful whole. Hence, λόγος is not only a word, but also a sentence, language as a whole, a reasoning, a discourse, and even reason. Thus, the profound ambiguity of man's position consists in the fact that his *nature* is to be rational. If anything, this ambiguous stance has only increased since Darwin made it clear that we are more natural than we ever thought we were. The more we teach that we are issued from a long evolutionary process, the more we proclaim our status as *extraordinary* animals, endowed with reason.

If we return to the substantialist account of language, either as a collection of symbols, or as a system of sounds, it would seem rather obvious that it is falsified by the most elementary observation of language. Indeed, language appears in our experience primarily as a process, a term which, though it did occur in Greek and Latin terminology, never entered into philosophical discourse as a fundamental concept, except in the neo-Platonic doctrine ([Plo51]) where it was used to describe how Absolute Unity revealed itself in a procession of succeeding layers of multiplicity.

Far from being a quasi-substance, language appears only as a form of interaction, whether we speak it, write it, or read it. Moreover, this process cannot be separated from the process of learning the skill of in-

teracting linguistically with others or with ourselves. Though the skill is the very condition of the process of language, it nevertheless develops itself within the course of the process of linguistic interaction.

The basic reason for the discrepancy between the received view and the observed phenomena may be traced to the ontological perspective which gave rise to the substantialist view of the world. This perspective was born from what we call the 3-P dogma (after the three initials of, respectively, Pythagoras, Parmenides and Plato), according to which the really real which hides itself behind the phenomena is immutable and therefore static. This static view has been kept alive along two different, and mutually exclusive channels. According to the first of these channels the ultimate reality of substances is traced to their form; according to the second of these channels the ultimate reality of facts is traced to their formulas (e.g. structure).

In as much as language is a process, any attempt at an adequate analysis of that process must start from a rejection of the substantialist ontology, and a respect for the main features of processes. It will be argued that C.S. Peirce provided a framework which does justice to those features.

3 Process and categories

3.1 Peirce's Categories

In its mature form, Peirce's doctrine of categories ([Pei35]) states that all phenomena present three aspects which, though irreducible to one another, have a different degree of dependency. The aspect of firstness is the aspect in virtue of which each phenomenon has an absolutely novel *quality*, unrelated to anything whatever. The aspect of secondness is the aspect in virtue of which each phenomenon involves an *interaction*. The aspect of thirdness is the aspect in virtue of which each phenomenon involves some *habit* (lawfulness, reasonableness, meaning etc.). Though secondness cannot be reduced to firstness, it presupposes firstness, and, similarly, though thirdness cannot be reduced to either firstness or secondness, it presupposes both firstness (through secondness) and secondness. Conversely, the element of firstness remains a mere possible, unless it be actualised by some interaction; and the element of secondness remains brutal interaction unless it derives its meaning from thirdness. The important point here is that the categories are related to each other according to a relation of *subservience*. The paradox of this relationship is

that, though thirdness (which is more complex than either secondness or firstness) nevertheless *needs* the relatively lesser significance of firstness and secondness. Thus, the categories must always be considered not only in themselves, but as they are relative to one another. This is even the case for firstness which, by definition, is unrelated to anything else. But this unrelatedness makes it a pure potential which needs a relation to a second in order to be discerned as a first.

3.2 Events and processes

Events are interactions. In terms of Peirce's categories, they represent the category of secondness ([Deb98]). But the mere fact *that* something happens says nothing whatever about *what* happens. The latter aspect is the aspect of thirdness. *What* happens in an event requires that the event be embedded in a context of events which are related to each other. Such web of related events is what is called a *process.*

3.3 Symbols, events and processes

Language consists of symbols. According to Peirce, symbols, share with icons and indices the structure of a sign. Every sign is constituted by the triadic relation between the sign itself ([Pei35]), its object and its interpretant. Because signs are generated from signs, and in turn generate other signs, every sign must be an event. From this it follows that language consists of sign-events which, by virtue of their intepretants, are embedded within a process.

4 Language and ontological perspective

According to the received view, language is a tool for communication. Moreover, the tool is more or less ready made and is used by people according to some rules, some of which are semantic (use the right lexical items to adequately reach the goal intended) and some of which are syntactic (use the lexical items according to certain rules).

According to the process view of the world, language is a process rather than a tool used by people. Moreover, people and the things which constitute the world are abstractions from processes. Language as a process is generated by symbol-events which are themselves generated according to rules which, in Peircean terms, are habits evolving from interaction with other processes. Language processes involve both syntactic and semantic rules or habit.

Linguistic symbols may be considered as gesture-events within a process of interactive responses. Syntactic symbols may be considered from two angles: the messenger and the receiver. From the point of view of the messenger, a syntactic symbol is a gesture announcing other gestures to be generated in view of the interpretant of the entire unit of meaning (e.g. a sentence). From the point of view of the receiver, a syntactic symbol-event elicits an abduction regarding a range of possible subsequent symbol-events. Thus, the symbol-event 'the' elicits an indefinitely large field of possible subsequent symbol-events, but excludes, for instance, the possibility of the next symbol-event being 'slept'.

Though Peirce distinguished different kinds of symbol-events according to their semantic value into quali-signs, sin-signs, and legi-signs, he did not, at least not to our knowledge, pursue the analysis of the function of symbol-events according to their syntactic value. But there is no reason why this can not be done.

Indeed, from a syntactical point of view, symbol-events have a specific function, regardless of their semantic function. The syntactic value of the language symbols making up the unit of meaning may be seen in function of the value which they have in forming such a unit.

By virtue of their secondness, events are marked by a binary relation which must be reflected in symbol-events. This is why, strictly speaking, one lexical item by itself has no meaning. The syntactical value of symbol-events will therefore depend upon the sort of relation that obtains between two language symbols. If one of the symbols has by itself no information content and therefore is a mere quality (a sound or a visible character), it will need another symbol to actualise its 'potential' content. Such nexus of two symbols, one of which has mere potential content, may be called a *proto-symbol* which corresponds to the category of firstness.

Similarly, when the nexus is constituted by an asymmetrical relation between one language symbol which derives its full content from its association with another language symbol which is in principle self-sufficient, it may be called a *deutero-symbol* which corresponds to the category of secondness.

Finally, when the nexus consists of a number of language symbols which are self-sufficient but together generate the interpretant of the unit formed by the string, e.g. a sentence, it will be called a *trito-symbol* which, by its aspect of thirdness, mediates between the language symbols constituting a unit of meaning.

To complete the picture, it is necessary to say a word about the *triadic relation* characterising each of these signs, because without such relation,

they would not be signs, let alone syntactical signs. But precisely what makes then *syntactical* signs is the very fact that they stand for specific *rules* or habits. Thus, the object of syntactical signs is the rule for which they stand. Their interpretant on the other hand is the generation of the selection of the next symbol-event. The interpretant of the entire string of language symbols is, from a syntactical point of view, the establishment of the correctness of the string, regardless of its semantic content.

Because syntactical symbol-events stand for rules governing the relationship between two or more language symbols, the occurrence of one symbol will trigger the expectation of the occurrence of the other relevant symbol as required by the rule in question. Such expectation which may be regarded as the expression of a hypothesis (which according to Peirce is the conclusion of an abductive inference) regarding the nature of the next language symbol, must obey Peirce's principle of abduction ([Pei35]). One of these may be called the principle of economy, which stipulates that all things being equal, a more plausible hypothesis must be tested before a less plausible one. Thus, the expectation triggered by a language symbol will be focused first on the occurrence of the other relevant symbol as required by the rule in question. The principle of economy is a very important factor in the construction of well-formed strings.

5 A process semeiotic view of language

NLCA defines three relation schemes underlying hierarchy (e.g. phrase structure) in language: qualification, minor predication and major predication. These relation schemes concretely illustrate the Peircean approach and become intelligible within the Peircean perspective. Qualification, minor predication and major predication are *implementations* of proto-symbol, deutero-symbol and trito-symbol, respectively. The process semeiotic approach of NLCA is also the basis of its flexibility ([KS98]). This aspect of NLCA, however, is beyond the scope of the paper, due to space.

5.1 Relation schemes

A *qualification* (Q) consists of a qualifier and a core. The qualifier has no information content independent of the core; it makes the core more specific. For example, a Q-relation (an instance of a relation scheme is called a *relation*) can specify the referential status of NPs; tense/aspect in VPs. The potential content of the qualifier is modelled by a *Proto-item* introduced for the core (in the case the qualifier precedes the core).

When the core is realized, it replaces the Proto-item. In the relation between qualifier and core, the qualifier points at some qualification of the core, and the core fulfils the combinatorial need of the qualifier. E.g. the article–noun relation.

A *minor predication* (*mp*) consists of a (minor) predicate and an argument. The predicate has information content independent of its argument and adds new, factual information to it. The relation between minor predicate and argument is asymmetric: the predicate needs its argument, but not the other way round (modification). The predicate points at some property of its argument, and the argument fills the combinatorial need of the predicate. E.g. the adjective–noun relation.

A *major predication* (*MP*) consists of a predicate and its argument(s). Both have information content, and the relation between predicate and argument(s) is symmetric (each requires the presence of the other). The predicate introduces an argument structure, and incorporates its arguments into a single relation. E.g. the verb–argument(s) relation.

The relations can be realised in language on different levels, e.g. on the morphological level, or on the level of syntax. This implies that NLCA has the potential to be language independent. At present we have a detailed specification of English. An NLCA application of Hungarian is being developed, and preliminary results indicate that our approach can also be applied to that language. In this paper we will restrict ourselves to English.

6 Evaluation principles

In NLCA the input string is analysed from left to right, and the relations are evaluated incrementally. A relation is evaluated when qualifier and core, or predicate and argument(s) bind to each other. The evaluation, which can be initiated by either participant in the relation, is economic, meaning that lexical items relate to the 'nearest' surrounding candidates.

This principle, called *greedy binding*, is restricted by the demand that only visible items can bind to each other. The visibility structure and any change to it, is due to the relations: each may introduce a new visibility range for itself. The creation of a new range makes an older range invisible, but when the new range ceases to exist, the older range becomes visible again. The hierarchy of ranges reflects the relation of subservience. In English, *mp*-relations do not change visibility, and this coincides with the optional character of modifiers (minor predicates).

A visibility range is terminated by *closing* (and by encountering end-

of-sentence). This operation applied to a combination of lexical items can yield a single new item, called a *lexical unit*. (N.B. a lexical item is a lexical unit; the principles described above extend naturally from lexical items to lexical units.) The linguistic properties of a lexical unit are derived from its members. Closing can be considered as an implementation of a property of signs that signs are generated from signs.

The object of a symbol-event is a rule. Whenever the rule is known, any constituent of the symbol can be used to represent it. This implies that in the case of closing no new item needs to be introduced, as long as there is no danger of ambiguity. E.g. in the case of a *Q*-relation, either of the qualifier, or of the core can represent the relation itself. We will make extensive use this feature in the example, in Sect. 7.

Finally, we say, the input is *well-formed* if the combinatorial need of each lexical unit is satisfied, meaning that the external argument positions of all items are filled (see below).

7 Towards an algorithm

Relations are represented by pointers between lexical units (in the case of morphological realisation, by constants). The source of a pointer is a lexical unit; the destination is an *argument position* of the related item. There are two *internal argument positions* for each lexical unit, representing information about the item itself: one for the *Q*- and one for the *mp*-relations, denoted by _int(q) and _int(m), respectively.

Each lexical unit has one or more *external argument positions* _ext, representing its combinatorial properties. In the case of verbs there are as many of such positions as there are obligatory arguments. The argument positions can be labelled, e.g. in the case of verbs; the labelling is defined by the lexicon, e.g. AGENT.

We represent the web of relations by a *Relation Matrix* (RM). There is a row allocated for each noun (a typical argument), called an object, and a column for each article, preposition, adjective, adverb and verb (typical functors), called attributes. Furthermore a column is allocated for each external argument position of a verb. For Proto-items a row or a column is allocated (referred to as Proto-object and Proto-attribute), depending on the qualifier introducing it. Lexical units created by closing are represented similarly, depending on the properties of their members. Internal and external argument positions of lexical items are given as buckets on the left and right hand side, respectively. Unfilled argument positions are omitted.

The pointers between rows and columns are (conceptually) stored in the cells of the RM; in the examples however, they are graphically represented. Besides the pointers, a cell contains a '+' sign if the destination of a pointer stored in the cell is an external argument position. These signs will be used as markers of the emerging phrasal structure.

All information about lexical units is contained in the lexicon, such as type, number and location (pre- or post-) of argument(s) etc. In the example below, lexical information is specified on-the-fly. The user of an NLCA implementation only needs to know about the relation schemes and NLCA's principles, no knowledge about the algorithm or the implementation is required. This implies that in such a model of language, information about lexical units can be expressed in terms of familiar linguistic notions.

Example *The lion slept in a cage.*

the generates a column. As a qualifier, it functions as the internal argument of its core, an object. The qualifier precedes the core, therefore it creates a Proto-object and points at its qualifying internal argument position.

the → Proto-object_int(q)

lion replaces the Proto-object. There is a qualifier present, hence there is a pointer from the noun to the external argument position of 'the', which results in a '+' in the RM in cell lion/the. We found a noun phrase. Since there is not yet a pointer to the external argument position of the noun, we still do not have a clause.

lion → the_ext	'+' in RM in cell lion/the

slept generates a column. Its qualifying internal argument is filled by the feature PAST; its external argument is labelled EXP. Since 'sleep' is a major predicate, it fills the external argument position of the object 'lion', and 'lion' points to the EXP role. As a result, there is a '+' in the RM in cells lion/EXP and lion/sleep. As the external argument position of the intransitive verb is filled, we found a clause.

sleep → lion_ext	'+' in RM in cell lion/EXP
lion → sleep_ext (EXP)	'+' in RM in cell lion/sleep

in generates a column. It has an optional first, and an obligatory second argument. As a preposition, it invokes a *MP*-relation for its argument(s), but when the relation is evaluated, the prepositional phrase

(PP) invokes a *mp*-relation for the lexical unit it modifies (Kamphuis and Grootjen, personal communication, 1997). In the current example, the first argument is not realised, the second argument has not yet been encountered.

a generates a column, and a new Proto-object (cf. the treatment of 'the' above).

a → Proto-object_int(q)

cage replaces the Proto-object (cf. the treatment of 'lion' above). We found a noun phrase which fills the external argument position of 'in'; as a result, we found a prepositional phrase.

cage → a_ext	'+' in RM in cell cage/a
cage → in_ext	'+' in RM in cell cage/in
in → cage_ext	

No new item needs to be introduced for the PP, as the preposition can also represent the phrase itself. In the example, we introduce a new external argument position of 'in', labelled PP, for representing the combinatorial needs of the PP.

The PP forms a *mp*-relation with the verb. The object of the symbol-event, 'slept in a cage', is the rule: there is a *mp*-relation between the constituents 'sleep(PAST)' and 'in a cage'. The recursive application of the triadic relation to the constituent 'in a cage', yields the rule: there is a *MP*-relation between 'in' and 'a cage'. The transitive closure of these relations yields the relation between 'sleep(PAST)' and 'a cage'; this results in a '+' in the RM in cell cage/sleep(PAST).

NLCA exploits the triadic relation for translating a relation between two columns of the RM, to one between row and column. Such a translation may be necessary when a RM is to be represented by a concept lattice (cf. Sect. 8).

in → sleep_int(m)	'+' in RM in cell cage/sleep
sleep → in_ext (PP)	

The Relation Matrix for the full sentence is displayed in Fig. 1.

Going through the sentence from left to right, we see the following structure emerge:

- At word 'lion' we obtain the noun phrase ($+_1$);

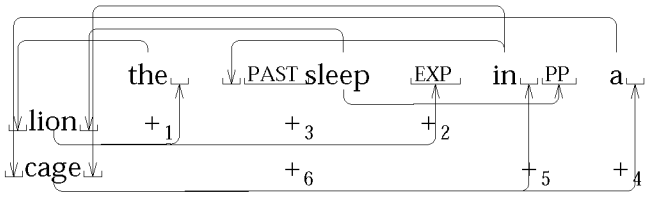

Figure 1: *The Relation Matrix for "The lion slept in a cage".*

- At word 'slept' we obtain the clause ($+_2$ and $+_3$);
- At word 'cage' we obtain the noun phrase ($+_4$), the prepositional phrase ($+_5$), and the full clause ($+_6$).

8 Conceptual structures

A RM can be represented as a conceptual graph (CG) ([Sow98]). Such a graph may be considered as a flow-chart of the symbol-events, in terms of their expectation-value as, respectively, proto-, deutero- and trito-symbols.

For each RM, there exists an equivalent conceptual graph. We verify this statement by defining a mapping between RM and CG, as follows. A lexical unit (l) is mapped to a box labelled l. If two lexical units are related, then there is a pointer between the corresponding boxes, and the pointer contains a labelled circle. The direction of the pointer and the label of the circle is defined by the relation. E.g. in the case of a Q-relation between qualifier and core, the pointer points from qualifier to core, and the circle is labelled 'q'. This pointer is the image of the pointers qualifier $\rightarrow$ core_int(q) and core $\rightarrow$ qualifer_ext.

The mapping of a *mp*-relation can be defined similarly. In the case of a *MP*-relation the mapping makes use of the labels of the external argument positions of the major predicate. For the yield of a closing operation a context is introduced. The mapping applied to the RM above is depicted in Fig. 2 (contexts are omitted).

A mapping from a RM to a concept lattice (CL) ([Wil82]) can be defined easily, too. It merely involves the removal of pointers in each cell of the RM. This may result in a loss of information, and that implies that the CL may only represent a RM in some respect.

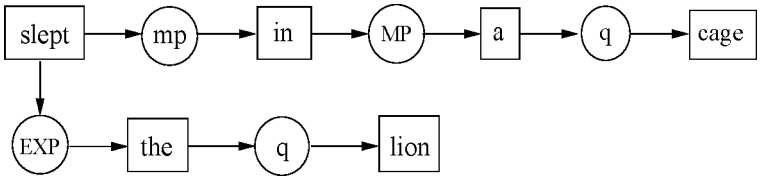

Figure 2: *A conceptual graph for "The lion slept in a cage".*

The CL for our example is shown in Fig. 3. The concepts of the lattice correspond to relations of the RM, and interestingly, to the focus of WH-questions: (C1) Who slept? (C2) Where ...? and (C3) What happened? This suggests a potential correspondence between question formation in language, and the information reflected in formal concepts. A (sub)lattice also has information content, e.g. C0-C1-C2-C3 represents the clause. The use of sublattices is potentially relevant for interpreting discourse relations.

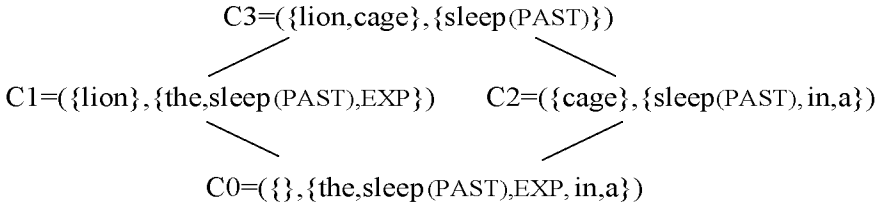

Figure 3: *The concept lattice for "The lion slept in a cage".*

9 Summary

A paradigmatic new model of language, NLCA, is described. It is shown that NLCA is consistent with C.S. Peirce's pragmatic, evolutionary and semeiotic approach, and that such approach supports and clarifies NLCA. Algorithmic aspects of NLCA are exemplified. The representation of an analysis by conceptual graph and concept lattice is discussed.

Acknowledgements

We thank John Sowa for providing a preprint of his forthcoming book, and we acknowledge valuable correspondence with Rudolf Wille.

References

[Ari84a] Aristotle. Categories. In J. Barnes, editor, *Complete Works of Aristotle*. Princeton University Press, 1984.

[Ari84b] Aristotle. Politics. In J. Barnes, editor, *Complete Works of Aristotle*. Princeton University Press, 1984.

[Deb98] G.J.Y. Debrock. Peirce's categories and the Importance of Secondness. In J. van Brakel and M. van Heerden, editors, *Proc. of the Int. Symp. on Peirce, C.S. Peirce: Categories to Constantinaple*, Leuven, Belgium, Leuven University Press, 1998.

[Hae91] L. Haegeman. *Introduction to Government and Binding Theory.* Basil Blackwell, Inc., Cambridge, MA, 1991.

[KS98] V. Kamphuis and J. Sarbo. Natural Language Concept Analysis. In D.M.W. Powers, editor, *Proc. of NeMLaP3/CoNLL98: Int. Conf. on New Methods in Language Processing and Computational Natural Language Learning, ACL*, pages 205–214, Sydney, Australia, 1998.

[Lam88] J. Lambek. Categorial and Categorical Grammars. In R.T. Oehrle, E. Bach, and D. Wheeler, editors, *Categorial Grammars and Natural Language Structures*, Dordrecht-Boston, D. Reidel Publishing Company, 1988.

[Pei35] C.S. Peirce. *Collected Papers of Charles Sanders Peirce.* Harvard University Press, Cambridge, 1931–35.

[Plo51] Plotinos. *The Enneades.* Desclee de Brouwer, Paris, 1951.

[PS94] C. Pollard and I.A. Sag. *Head-driven Phrase Structure Grammar.* The University of Chicago Press, Cambridge, MA, 1994.

[Sow98] J.F. Sowa. *Knowledge Representation: Logical, Philosophical, and Computational Foundations.* PWS Publishing Company, (forthcoming), Boston, 1998.

[Whi85] A.N. Whitehead. *Science and the Modern World.* Free Association Books, London, 1985.

[Wil82] R. Wille. Restructuring lattice theory: An approach based on hierarchies of concepts. In I. Rival, editor, *Ordered sets*, pages 445–470, D. Reidel Publishing Company, Dordrecht-Boston, 1982.