EPRINTED FROM:

ournal of
PRAGMATICS

Interdisciplinary Monthly of
nguage Studies

rnal of Pragmatics 20 (1993) 467–492
th-Holland

ragmatics, cognitive science and connectionism

ter Harder and Ole Togeby*

eived December 1990; revised version September 1992

sevier Science Publishers B.V. (North-Holland)
nsterdam · London · New York · Tokyo

Journal of Pragmatics 20 (1993) 467–492
North-Holland

Pragmatics, cognitive science and connectionism

Peter Harder and Ole Togeby*

Received December 1990; revised version September 1992

The article looks at two types of computer simulation of mental process: the so-called 'classical', serial architecture based on discrete input categories and a compositional syntax and semantics, and the 'neural network' approach based on connections that determine input–output relations. After giving an outline of the main features of the two types of architecture, we discuss the claims made on their behalf, and try to sum up their strong and weak points as candidates for duplicating human processes of understanding. A key point in the argument is the discussion of the 'recognition of intention' aspect of human understanding, where we argue that computer simulation can only handle intentions in terms of plans that can be taken as fixed in advance, while human intentions can only be understood as based on an assumption of freedom of action, implying notions like responsibility and free will. Having suggested how different aspects of language understanding fit into this picture, we conclude that no matter how far computer simulation proceeds, the inherent discrepancy between the status we attribute to a human subject (as part of our basic pragmatic competence as fellow subjects) and the status we attribute to a computational process (as part of an inherently controllable human plan) makes it contradictory to assume that a full understanding can take place by simulation. The contradiction, however, is not theoretical: it emerges from the ground rules of human practice.

1. Introduction

In the following we are going to take up some issues that are currently being discussed in cognitive science, with the main emphasis on the confrontation between 'classical' artificial intelligence based on serial processing and symbol manipulation, and accounts in terms of parallel distributed processing, based on 'neural networks' organized in terms of nodes with connections determining intput–output relations – from which stems the term connectionism. The first approach will for convenience be abbreviated AI, although many

* *Correspondence to:* P. Harder, Department of English, University of Copenhagen, Njalsgade 84, DK-2300 Copenhagen S, Denmark. E-mail: harder@anne.ihi.ku.dk

We would like to thank Jan Riis Flor, Frans Gregersen, Philip Johnson-Laird, Hanne Ruus and Ib Ulbaek (plus our two anonymous reviewers), who patiently pointed out errors and exaggerations in earlier versions of this paper. Needless to say, they are not responsible for imperfections in this version.

0378-2166/93/$06.00

would think of AI as covering the entire field, and the latter will be referred to as PDP, although there are other types of PDP than connectionism. We shall argue that in order to see these issues in a proper perspective, a pragmatic dimension is essential, and that an awareness of precisely where pragmatics comes in will make it possible to set up a division of labour between the two types of cognitive architecture, where neither of them has the last word, but both offer illuminating insights.

Although we occasionally digress into other fields the point we would like to make is basically about natural language understanding. As can be seen in the themes of the 1990 and 1993 pragmatics congresses in Barcelona and Kobe, the relationship between cognitive and social aspects of language is one of the challenges that research in language is currently trying to face. That language crucially involves cognitive aspects has been generally recognized after the heyday of behaviourism; that language has crucial social aspects has been generally recognized after Wittgenstein – but it remains controversial to what extent the 'social interaction' aspect of language needs to be taken into consideration in accounting for the arguably central cognitive aspects. Our position is based on the assumption that cognition is basically a pragmatic activity, interwoven with the social process, and the feeling that the status of cognitive processes as an aspect of human activity is not sufficiently recognized in discussions on cognition, in which a form of intellectualism is often one of the basic assumptions.

2. AI and PDP

First of all, it should be made clear that although we may appear to make sweeping statements about the entire state of the art of computational description, what we are trying to do is much less ambitious. Our aim is to talk about two fairly general types of simulation, which we take to be important enough to be worth discussing without attempting to give anything like an exhaustive impression of the work that is presently going on in the field. What is important to us is the basic orientation of each of the two approaches we focus on; we do not claim to cover everything.

Before we go into the differences between the two approaches, there is a similarity which is crucial from a pragmatic perspective. Both positions arose as part of the same research paradigm: the attempt to simulate mental performance on the computer. Certain aspects of the basic research activity have therefore necessarily shaped the way both camps think about cognition. Progress in research is necessarily progress in terms of what the machine can do; what happens outside the machine is outside the focus of interest. It is more or less inevitable that getting the machine to solve a problem in one or the other way will steal some of the attention from the job

of getting a plausible complete picture of how cognition as a human activity works.

This does not mean that we accuse practitioners of computer simulation of lack of interest in the real world; on the contrary, simulation of ways in which text processing depends on world knowledge is becoming subtler every day. Our point concerns the difference between the situation of a 'machine-in-the-world' and a 'human-being-in-the world'.

We do not want to take up a 'holier-than-thou' attitude on this point, and we think there is every reason to be excited about the prospects of computer-based work on cognition; we also think this is wholly compatible with the attempt to be aware of the fundamental embeddedness of cognitive processes in human practice. We aim only to show how this basic pragmatic perspective, which tends to be overlooked, may serve to deflate certain exaggerated claims on behalf of machine 'architectures' of whatever kind.

This means that we have no quarrel whatever with exponents of AI or PDP who see what they do only as a way of *describing* the things that a human mind must be able to do, similar to ordinary language, but more precise and reliable. For instance, there may be very good reasons for describing human language activities in logical terms without thereby implying that human processing is a matter of logic. However, it has been a fairly widespead assumption that the programs are not just capable of yielding descriptions of human performance in terms decided by external considerations, but that the programs were actually capable of duplicating mental processes. This would imply that what goes on inside the machine (and not just the input–output relation, as seen from a particular theoretical perspective) matches up with actual mental processing. The very term 'artificial intelligence' would seem to raise this kind of expectation – but we think that it is a question which would be worth considering even if nobody understood computer-based work on cognition in this way. Our discussion is to be understood as an attempt to throw some light on the issue of similarities and differences between what goes on in the two architectures and actual human understanding.

3. Survey of 'classical AI'

In classical AI, simulation of human understanding and interpretation of texts is done by a program which consists of a dictionary, a grammar (dealing with syntax and semantics), and a parser.

The dictionary is a list of all minimal signs of the language that we want to be able to process. The list does not contain letters but only minimal signs, i.e. morphemes or words (fixed combinations of letters having a constant form-meaning relation).

The grammar is a set of rules that describe the sentences (or strings) of a language and the structures that underlie those strings. The rules of the grammar can be seen as the set of instructions used by the machine in deciding whether or not a particular sentence is part of the language. The grammar consists of a set of nonterminal symbols (names of constituents), a set of rules that rewrite a symbol as a string of symbols or words, and an axiom, namely the initial symbol 'S', which is rewritten, in accordance with the rules, in ways which correspond to any sentence that is part of the language described by the grammar.

The parser is a set of instructions on how the machine using the grammar and dictionary is to generate an analysis of an input string. It is a set of rules specifying the order of application of the rules of the grammar and instructions for the input and output devices to work. The parser does not describe the linguistic expression as such, but describes the process of analysing the linguistic input. A Markov process machine is one type of parser, an ATN (augmented transition network) is another, and a left corner parser a third type of parser. There are many others.

Normally an AI interpretation model takes as its input a text and yields as output an analysis of the text in the form of a kind of formal representation – tree diagrams, bracketed strings, logical expressions, translations of the meaning of the text into a feature structure, or a sort of markerese, an artificial language with well-defined properties. We shall be talking about tree diagrams, but our argument is based on the assumption that in the respects we are interested in, other representations would not be essentially different. If the AI model is taken as a cognitive model, not only the grammar is relevant, but also the dictionary and the parser – especially if it is compared to a PDP model.

An important feature of a classical grammar and parser is its seriality; it only takes one step at a time, and all steps made by the machine are ordered in a sequence. That means that options in the grammar are computed by backtracking. If the grammar allows a choice between, say, two word classes in one position, the machine first computes one option until the analysis is finished or fails, then backtracks to the point of choice, and then it computes the second option until the analysis is finished. In many cases with options in the grammar a serial machine will backtrack for a very long time, and overgenerate, i.e. create many possible solutions to the same task. Serial parsers can handle options in other ways. You can rule out some of the generated structures by means of a filter which (according to specified principles) excludes some of the generated structures, by a preference mechanism, by lookahead or by tables, but all have runtime problems of the same dimensions as those involved in backtracking.

The role of the grammar, the lexicon and the parser can differ in different AI approaches; there has been a tendency towards less grammar and more

information in the lexicon. But common to all AI aproaches is the principle of compositionality, namely that if you know the meaning of the elements (in the lexicon) and how the elements are combined (in the grammar), you can calculate the meaning of the whole. Whatever procedure you use, the processing presupposes discrete input categories.

In processing natural language many items will potentially belong to many such discrete input categories, as exemplified above. The discussion of backtracking and its alternatives was meant to illustrate the kind of problems this gives rise to. Whatever way is chosen to handle the problems, any system must have a mechanism, containing many computational steps, for filtering out the unwanted readings.

4. Survey of PDP

The PDP model consists of a set of input 'neurons' (processing units), a set of output neurons, a set of connections and hidden neurons which connect the input neurons with the output neurons, and a training set consisting of input strings, and correlated canonical output strings.

The input neurons are processors which represent – or, more precisely, the programmer uses to represent – distinctive features of the input string. Distinctive features can be (parts of) phonemes or letters of a word, dots of a picture, or semantic features of words and clauses. Every input neuron transmits a signal if it 'recognizes' the input feature. Output neurons are processors that transmit signals with defined meanings – or, more precisely, signals which the programmer uses to designate defined meanings. An output neuron signal can designate a concept, a gestalt type or a sentence type. In some cases if the input value to the output neuron exceeds a certain threshold value, the output neuron 'fires', i.e. transmits the signal, and if it does not exceed the threshold value, the output neuron does not produce its output signal. In other cases the output may vary as a (non-linear) function of the input, so that the influence on the output achieved through an increase in input depends on the overall size of the input (cf. Johnson-Laird, 1988: 186).

Connections are 'pipelines' which (in a common design of neural networks, viz. feed forward networks) connect the input neurons with the output neurons; they eighter strengthen or inhibit the signal depending on their weight or strength. The recent resurgence of interest in systems based on such processors is due to the addition of so-called hidden neurons, processors which are intermediate between the input and output layers and thus contribute an extra layer of connections, which considerably increases the power of the system.

The training set is a series of 'trial runs' used for adjusting the weights of the connections. PDP networks are 'trained' by a process which involves comparing the actual output with a target output, so that if actual output is off target, the connections responsible for the 'wrong' output are corrected by the process of back propagation. If the output is too strong the connections to that unit are weakened; if the output is too weak, the connections to that unit are strengthened. So the weights are 'trained' to deliver the desired output as a response to the prepared input. When the network is trained, it can process any kind of input and always gives some sort of output – not always the right one – chosen among the possible patterns of firing output neurons. What takes the place of rules, i.e. the network's 'knowledge' of regularities and uniformities in the input string, lies in the pattern of weights of the connections.

The input to a PDP network could be a (polysemous or ambiguous) word in its context, a picture or a sentence; and the output could be the meaning variant of the word, the name of the depicted thing, or the type or function of the sentence.

In principle, the name 'parallel distributed processing' means that all the connections transmit their signals at the same time, and all the neurons transmit their signals at the same time. That means that the processing is very fast. The back-propagation process during training, on the other hand, is very slow. This is not surprising in view of the fact that all weights are adjusted for each piece of training material; normally there are over 10,000 connections, and the network often needs hundreds of runs before it has 'learned' the correct input–output relations.

In practice most PDP networks are made in normal serial computers, and the parallel processing is only simulated by the ordering of the instructions to the computer. All the connections transmit their signals before the sum of the input signal is computed at the first output neuron, and so on. Often the grammatical insight built up in the pattern of weights in the PDP network is part of the same linguistic knowledge as the programmer puts into the serial computer program; so in a way a PDP program can be viewed as an alternative to the parser and to parts of the grammar, not an alternative to the grammar alone.

5. A comparison

Some of the interesting features of PDP accounts compared to classical AI are the following:

(i) The processing takes place at a sub-symbolic level, i.e. the input–output relationship is not mediated by any semantically interpretable symbols of the input. In a sense it is as when a picture is processed as a series of coloured dots on a tv-screen; the picture is not in any of the single dots, but in the pattern of all the actualized dots.

The relation between the sub-symbolic level, the connections, and the symbolic level, the output, in a PDP process can be compared to the relation between phonemes (letters) and signs in human communication. At the sub-symbolic level the minimal parts have no meaning, only distinctive function. Thus, one connection in a network has no meaning, either in the system (i.e. the architecture of connections), or in the processing; the meaning is mediated by the pattern of activated connections. Similarly, one phoneme in a word has no meaning, either in the system of phonemes, or in the actualization of a given word; the meaning is mediated by a pattern of actualized letters.

Writing systems with letters (representing phonemes) are much more economical than writing systems with logographs (representing words or morphemes as in Chinese writing) with respect to the inventory of signs. So a PDP network is more economical than a dictionary as a computing device as far as the processing of the input string is concerned, in a PDP network the same connections are used for the computing of many different pieces of meaning, while every piece of meaning must have its own dictionary entry in a normal parser.

A symbol manipulating program must have a dictionary containing all possible words or symbols in the natural language, while a connectionist program will do with a limited set of distinctive features to process the input string. It should be mentioned that the economy is not guaranteed, only a possibility: the number of distinctive features will be crucial for how succesful the processing is.

(ii) Because the processing takes place in a large network of interconnecting units, there is no well-defined place in the system where a particular aspect of the process always may be said to belong. All messages are processed globally in the system (although some areas in some cases will have specific meanings). This makes the process very robust. Noise, false signals, deleted or damaged signals of the input will not damage the output as a whole – exactly as single misprints do not spoil the meaning of a text. In a PDP network the error on the output will be proportional to the damage on the input signal ('graceful degradation') – in contrast to normal serial computer processes where the whole output is damaged or totally missing because of only one single misprint in the input (leading to the charge of 'brittleness').

(iii) In (feed forward) PDP networks the range of output signals is limited and predetermined, comprising only the number and types which the designer of the network wants and has foreseen. Conversely, the network can recognize almost any input, viz. all combinations of sub-symbolic signals, as belonging (or not belonging) to one of the already known output types. Classical serial computation, on the other hand, has a limited and predetermined range of input signals, consisting in the signs already written by the programmer in the dictionary – but allows an infinite number of output signals, viz. all complexes made by (recursive) combinations of input signs. There is therefore a potential

element of economy on the output side for AI, just as there was on the input side for PDP.

Put differently: PDP can yield recognition of complex input tokens as belonging to a type, but cannot generate complex output structures out of simple elements; classical computation cannot recognize complex input tokens as belonging to a type, but can generate or compose complex wholes out of simple elements.

(iv) PDP processing typically presupposes a phase of training in which the weights of the connections are adjusted. That means that the programming, the making of the rules in the grammar of the program, is made through interaction between the system and the environment, viz. through the back propagation process. Classical serial programming is made once and for all before the start of the input–output interaction of the system with the environment. In so-called learning algorithms the learning is the input–output interaction of the system, and the rules of learning are not changed during this interaction.

That means that the motor, the basis and the driving force of the PDP program is the evaluation of the output by the environment (the wishes, the desires of the programmer). Normally we say that an implemented Chomsky-grammar is syntax-driven, which means that the motor and the basis is the grammatical rules and the lexicon.

In a way the PDP programming may be seen as mirroring a scenario for the development of human cognitive capacity in childhood, with the programmer in the role of a social partner; similarly, serial computation may be seen as mirroring the innate capacities of human beings, where the programmer plays the role of the creative God. PDP programs are plastic and approximative, serial programs rigid and precise.

This summary has tried to make the most of the differences between the two types of architecture. Before we go on, we need to emphasize a central hedge about the distinction. As already mentioned, most PDP networks run on serial computers. Similarly, parallel machines could run serial processes. Also, in each architecture programmers may adopt strategies that make it very difficult to tell whether a given machine performance is based on one or the other kind of architecture (There are, however, serial processes which cannot be matched on a PDP network; more on that below). But when in the following we mention AI-*like* or PDP-*like* processes, we are not taking for granted any simple relationship between architecture and process type – we are talking about two (proto)types of mental process, with a view to suggesting a plausible place for them both.

6. The Argument. Compositionality and meaning

The feelings generated by the controversy are fuelled by a well-known ideological opposition: connectionism looks like a plausible imitation of some aspects of neurophysiological processes, or what one would traditionally call unconscious, automatic processes; the system is made to work through what is essentially a process of imitation: the machine is told what outputs it should produce and then works out a way of getting there – but as with a parrot, the process that leads to the output is not assumed to matter. As against that, classical AI with a level of logical syntax looks like a simulation of well-established notions of logical, consciously explicable, mental processes. To the extent that connectionism succeeds in accounting for performance that would otherwise be ascribed to logic-like processes, it may be seen as a setback for human dignity, conceding territory that had been reclaimed from behaviourism once again to mindless routines.

A lively example of this debate can be found in Fodor and Pylyshyn (1988) and Smolensky (1987), some features of which will be discussed in the following. The notion 'level of analysis' is central to the structure of the argument. The physical world is organized in many levels – particles, atoms, molecules, etc. – and phenomena due to effects at the molecular level cannot be analyzed by looking at particles. But saying that there is a molecular level does not, of course, amount to denying the existence of particles. Fodor and Pylyshyn, as champions of the AI position, argue that one can have the advantages of both positions by seeing PDP-type structures as representing a lower level in the hierarchy of mental process. What at the low level is messy and gradual, is at the higher level organized into precise units; PDP *implements* AI categories.

Smolensky, in his reply, puts himself in a dialogue-oriented position, by accepting one of the basic tenets of classical AI, viz. the existence of properties that are not gradual and messy in the cognitive process. Summing up the positions of AI and PDP in the slogan words 'hard' and 'soft' (respectively), he reviews several possibilities, dismissing attempts to deny the existence of either 'hard' or 'soft' properties, or of allowing them to exist in separate compartments of the brain. He also argues against having a basic level of 'hard' rules, which because of their complexity can have emergent 'soft' properties, and argues instead that the fundamental level is 'soft' but takes on some emergent 'hard' properties. This puts him, as he points out himself, quite close to the position of Fodor and Pylyshyn on certain points. The difference is that according to Fodor and Pylyshyn, once you go one level up, there's an end of softness, and we can go with AI the rest of the way. PDP belongs at the basement level, and should stay there. Smolensky, however, thinks that softness remains fundamental, but only takes on some hard properties, which are approximately like the hard categories of AI. So the hard and discrete categories of AI are not a higher-level truth, but only an approximation to the more complicated facts that PDP-structures, if amended as suggested by Smolensky, alone can account for.

A key argument in the discussion is the constituent structure of mental representations (cf. the title of Smolensky's paper), or, to put it differently, the combinatorial properties of elements in mental representations. Fodor and Pylyshyn argue (and Smolensky, by agreeing with them, locates himself on a 'middle ground') that it is necessary with an account that can identify elements across contexts, so that mental representations can be seen as having some constituent structure.

Smolensky gives as an example the case of 'cup with coffee' and 'cup without coffee'. In contrast to those who would want to see these complex concepts as completely unanalyzable wholes, Smolensky points out that you can in fact work out an analysis of 'coffee' by comparing the two. In contrast to those who would want to say that we therefore have a 'hard' level of discrete symbols, Smolensky points out that the representation of coffee we get out of the comparison is a very special kind. It is precisely the kind of coffee that is relevant in combination with the word 'cup'. One of the features involved is therefore e.g. 'hot liquid contacting porcelain'. Smolensky uses this to argue that symbol meanings are not discrete, precise and identical across contexts – they involve, rather, family resemblances of the gradual kind that PDP captures.

But does this answer all the objections that Fodor and Pylyshyn make against letting softness have the last word? As far as we can see, there is one argument that has not been answered. What Smolensky shows is how constituent structure in some restricted cases can be preserved, even if complex constructions are analyzed, as wholes, in terms of PDP – and how this can be done without handing over everything to a superimposed 'hard' level. But the central objection of the Fodor/Pylyshyn article is that the notion of analysing all complex constructions as belonging to pre-established categories is basically implausible. Their key argument is that the number of twenty-word English sentences is somewhere in the order of the number of seconds since the big bang. If the brain were to treat such sentences in PDP-fashion, as described in the last section, it would require a pre-established inventory of output categories comprising this fantastic number of possibilities.

But the difficulty is not only numerical. It is also a matter of whether understanding an utterance is basically a process of putting it in the right slot – which is essentially what a PDP network does. Neural networks do not 'understand' utterances or pictures – they divide them into categories after having been trained to do so, cf. the description in section 2. If understanding of linguistic utterances was like that, it would be like guessing which of father's twelve funny stories he was in the process of telling right now; understanding consists in assigning the utterance in progress to a category, e.g. 'joke No. 6'. The important truth about compositionality is not only the ability to recognize constituent elements – it is also the ability to use elements to build up entirely new combinations, i.e. combinations which are not instantiations of a pre-established type consisting of all such combinations. Understanding is more than recognition.

So the Fodor-Pylyshyn objections to a mental architecture based entirely on PDP remain in force even if we accept that a Smolensky-type PDP could show emergent 'hard' properties. However, we think that the facts they point out should be understood differently from the way they are understood in the AI view. At the heart of classical computational processes lies the notion of an algorithm, an 'effective' procedure that is fully explicit and decomposable into minimal steps of the same fundamental kind that enter into the original conception of the Turing machine. Corresponding to this one must assume that mental representations, as involved in the production and reception of natural language, work according to a fully compositional, fully explicit procedure that works out the meaning of the whole by working out the meaning of the parts.

What is wrong with this view can be illustrated with reference to the notion of the 'hermeneutic circle', which (in Schleiermacher's version) says that you cannot understand the whole before you understand the parts, and you cannot understand the parts before you understand the whole. The reason for this is the fact that the meaning of an ambiguous part is not clear before you know the meaning of the whole in relation to its context. The hermeneutic circle means that you cannot reach an understanding by putting constituents together, as in an algorithmic procedure: there are aspects of understanding that are not cumulative. So just as we had to reject the PDP-oriented process of assigning the utterance to one among an inventory of pre-established output categories, on a hermeneutic view of understanding we have to reject the AI-oriented view based on computational operations working from pre-established and unchanging input elements.

The basic point in showing why this is so has to do with the nature of the input to compositional calculation. Pre-established input elements are a necessary condition for calculability, but the establishing of the input elements themselves is not part of the calculation; and the hermeneutic circle implies that we cannot establish the precise nature of the contribution of each part independently of the process of working out the meaning of the whole. The parts, necessary as they are, are not the whole story about how to make sense of a complex whole. This does not mean that we contest the validity of the Church-Turing thesis. We do not doubt that what can be calculated by humans can be calculated by an explicit algorithm too; what we doubt is that the meaning communicated by language, i.e. the semantics of a whole utterance, is exhaustively describable in terms of calculation. Putting two and two together is necessary, but it is just not enough. Neither input nor output occur as purely discrete well-defined units on which the effective procedure can be used. Understanding is more than cumulative composition (cf. also Penrose, 1989, on the non-algorithmic aspect of mathematical understanding).

7. Intent-driven cognition

The picture that we are going to offer involves concessions to both classical AI and PDP. Recognition and combination are both essential; overall categorisation of aggregates and identity of constituents are both parts of the picture. But an account in terms of levels or an account in terms of emergent hardness is not satisfactory, because the bottom line of understanding has to do with the social process, not with either discrete elements or pre-established output categories.

To take the example of 'coffee': Smolensky is right in pointing out that assigning a meaning to 'coffee' involves finding the right place in a loose set of family resemblance, not simply an assignment to a discrete unit in the system. The word means something slightly different in 'can with coffee' and 'cup with coffee'; and this problem cannot be fixed by having a function that modifies the input meaning for each element in the *syntactic* context (cf. Lewis, 1972), because determination of actual token meaning (interpretation) ultimately involves the communicative situation. Examples like 'the ham sandwich is sitting at table No. Five and is getting impatient' (cf. Nunberg, 1978, as quoted by Johnson-Laird, 1983: 241) involves assigning meaning to 'the ham sandwich' based on what makes sense in the situation, rather than simply syntactic combinations (more on this in the next section).

But Fodor and Pylyshyn are right in saying that PDP can only handle this if we know in advance all the possible variations there may be. The whole field of family resemblances must be ready for us to recognize an invidual occurrence of the word as belonging to one particular corner of the field. If we combine elements in such a way that understanding requires a creative effort, we put PDP out of business. A network trained to recognize pictures of giraffes from pictures of camels does not have the option of saying: Ah! A giraffe with two humps! – it has to stay within the pre-established categories. Either it is a giraffe or it is not.

The capacities of PDP in relation to creative processes could be illustrated by the results of applying the technique in the field of computer music. Loy (1989a,b) claims that there are a number of things about music that offer exciting prospects for PDP. Learning is often achieved through mimicking ('do like I do!') rather than through precise prescriptions; gradual variation plays an important role, and musical skill involves an ability to maintain one's orientation across a spectrum of variation that fits badly with the 'brittleness' of AI, etc. (but cf. Johnson-Laird (1988: 260ff.) on non-PDP-like aspects of musical competence). However, what the networks can do is always to identify the already established invariant elements across different contexts. It can do this in ways that are subtle and many-faceted; for instance, a network may pick up patterns of tone sequences that correspond to general, cultural patterns, and also specific patterns that identify a particular piece of music

(cf. Bharucha and Todd, 1989). As soon as a type has been established, PDP techniques may pick it up, and by identifying a specific sequence and simultaneously putting it outside the cultural norm, it may even identify something which is a creative departure from the norm. But this is done by combining two types of pattern identification; and it is only the pattern identifications that are performed by the machine. It is the human observer who identifies the creative element in the discrepancy between two established patterns – or who, as a composer, decides that just this tone is the right one to put next.

Before we go on to suggest our own views, we can sum up an important part of the discussion so far by saying that PDP because of its fixed output categories is well adapted to account for 'backward-oriented' aspects of understanding: everything about the utterance that must be understood by assigning it to pre-existing categories, regardless of deviations. On the other hand, AI because of its fixed element categories is well adapted to account for some of the 'forward-oriented' aspects of understanding – those that can be captured in terms of shuffling the cards in a new way.

As already announced, what we see as missing in the pictures of cognition offered in the debate referred to above is a pragmatic understanding of the nature of cognitive process. The 'forward-oriented' aspect of understanding that is missing from the AI picture is the fundamental forward-orientation that is implicit in the notion of action itself – the fact that an act is an attempt to bring about an intended purpose, as pointed out by Grice and followers. The crucial feature of intentions in this context is that an intention is a step into the concrete, pragmatic future of the speaker and audience. Understanding involves knowing to what future point the utterance is designed to move the interlocutors. All aspects of cognition involved in understanding is subordinate to this fundamental task. What we talk about here is not the rules relating means and ends, but the very choice of goals of a given interaction between speaker and addressee in an ongoing situation. As shown by Shank (1977), once you have a scenario in which the goals and intentions can be taken as settled, intentions are not problematic any more; the problem is that neither goal nor subgoals can be assigned definitively in advance of understanding any given utterance.

That this is so for ironic and metaphorical communication has long been generally recognized, but the notion of 'literal meaning' has made it possible to preserve the idea that 'regular' communication could be dealt with on AI terms, in the absence of the notion of communicative intention. There are several possible objections against this view. Ambiguity or multiplicity of meaning is such a ubiquitous feature of everyday language that a combinatorial explosion would put most known procedures relying on a series of yes-and-no decisions in a syntactic tree or on yes-and-no filter devices out of business very quickly as a plausible account of mental process.

The word *pass*, to which we return later, can mean 'go past' (*pass the bridge*), 'come to an end' (*the moment passed*), 'reach the required standard' (*pass the exam*), 'approve' (*pass the law*), 'make no bid' (*after four hearts, his partner passed*), 'hand over (*pass the salt*), 'be tolerated' (*I'll let it pass*), not counting idiomatic combinations like *pass water*. This situation is typical for frequent words, and is especially rampant in the case of prepositions, which tend to have more than ten clearly distinct readings. In a normal everyday sentence, which is immediately intelligible in context, there can be a series of highly ambiguous items which the processor must decide on in order to be able to make inferences.

In a normal serial computer program this series of decisions would be made by backtracking and it would be impossible to simulate it on a computer 'in real time'. It would take minutes or half hours to accomplish even small disambiguation tasks. And even if a serial procedure could be constructed which would do the trick, the 'leave-no-stone-unturned' aspect of effective procedures makes them very ineffective and hence implausible as candidates for resolution of ambiguity (compared with the 'jumping-to-conclusions' aspect of PDP). Even in AI short cuts, all steps must be fully specified.

But this practical problem should only be seen as a pointer towards a more principled objection to the view of meaning that is embodied in the algorithmic view of interpretation – the reliance on overall intention in fixing the meaning of the parts on which a calculation would have to be based. That is not to say that intentions could not be calculated by a program. In fact it is possible to calculate ironic interpretations provided we have the sentence *and* the ironic intention – but the problem is that the intention has to be part of the input to the computation. We cannot generate intentions from linguistic input. As we saw above, a word like 'coffee' – even in fairly pedestrian combinations – offers a spectrum of interpretations united by no more than family resemblances. Representations must change, not only as a result of syntactic operations, but also as a result of what fits the intention of the speaker. If Joe wants coffee, he may want something to drink; if Nancy wants coffee, she may want something for granny's visit tomorrow. These everyday cases have the same feature as the 'ham sandwich' case – the syntactic context alone does not provide all the information we need.

As a first test for 'intent-driven' cognition we may return to the question raised by our rejection of classical AI as a complete theory of utterance understanding. How can we preserve the element of compositionality that makes it possible to avoid having a number of output categories similar to the number of seconds since the big bang? The answer involves an adjustment in our understanding of linguistic meaning.

8. Instructional semantics

The slogan 'instructional semantics' embodies our version of a view of meaning wich in different versions have been advanced by procedural semantics (cf. e.g. Winograd, 1976; Johnson-Laird, 1977), discourse representation theory (Kamp, 1981) and relevance theory (Sperber and Wilson, 1986). The essential aspect is to see meaning as input to the process of interpretation. Meaning, according to this picture, is something that affects the audience's state of information – or, more generally, their overall position with respect to the speaker and other people in the situation. It needs to be emphasized that information is too narrow as a cover term for what linguistic meaning does to the audience.

According to this picture, the meaning of a lexical item cannot be specified in advance in all possible contexts, as assumed by AI. Neither is there a pattern of firings that uniquely determine the range of possible inputs that would have a particular lexical item as output, as would be necessary if PDP were to cover lexical meaning adequately. Rather, a lexical item (as used in a given utterance) is a signal to the audience to apply the meaning potential for a certain word in a way that is consistent with their overall interpretation of the intention of the utterance of which it forms part. There is neither a fixed meaning, nor a predetermined pattern of variation – only the instruction to the audience to use what they know in a way that makes the best possible sense.

The meaning potential of a word we understand in such a way that it has properties reminiscent of neural networks, as well as of traditional descriptions of memory organization; cf. also the relations between meanings of polysemous items in cognitive grammar (Langacker, 1987, 1991; Lakoff, 1987). It involves a number of nodes that are more or less closely associated, and connected along many different paths (cf. Johnson-Laird, 1988: 328). All properties, both of form and meaning, are multiply organized in a net of associations, so that activity is widely distributed in the associative net when understanding takes place. What particular type of organization must be assumed for a given word depends on a number of factors, but in order for a word to survive and be useful, it must of course be organized in such a way that communication about the sort of things that people in the speech community actually talk about is helped by the existence of that particular word – or speakers would not learn it. The word 'cow' presumably bears a privileged relation to experiences and knowledge of members of a certain natural kind; also, the word is associated with milk, possibly with calves and animals in a field saying 'moo', etc. The central status of the natural kind means that the natural kind would be directly related to practically all nodes in the 'cow' net, while other nodes might not be related except via that connection (e.g. 'cow/bull' and 'cow/milk')

This net of associations could be seen as a common data-base for knowledge about the world and knowledge about language. The difference between the two would be basically determined by the difference between individual and social aspects of the 'cow' net. For speakers who had a long and happy record of riding on cows, there would be a close association between cows and riding; but if they were competent speakers of the language, they would not use the word in 'riding' contexts, apart from conversations between old friends, because as speakers they would be able to distinguish the communicative potential of the word from its private associations. Thus, the use of the word for communicative purpose determines a certain search strategy, different from that of reminiscing about cows. Similarly, if we search the 'cow' net of associations for the purpose of finding information about cows that is relevant for a particular context, we organize our search differently again. Lexical items are crucially social phenomena; their *raison d'être* is that if we let our welter of experience and knowledge be organized by principles that have social validity, we can draw on these resources together.

What this means is that what makes a lexical item identical across contexts is the entry to a net of associations connected with that particular lexical item, not what communicators find within those associations. Compositionality exists at the level of potential meaning; but it is the actual meaning, i.e. the interpretation that involves representation. The context, linguistic as well as situational, co-determines where you are going to end up. And you may end up with a variant that you have never come across before.

If the meaning of a sentence does not in itself add up to a representation of a state of affairs, this raises the question of how speakers are able to represent to themselves the interpretations of utterances that they work out. Fodor's assumption that there is a language of thought which is structurally close to the human language that we speak must stand in suspicion of needless duplication of the notion 'language'. If, with Johnson-Laird (1983), one instead assumes the ability to build 'models' of states of affairs as a general cognitive skill, one can say that the model-building capacity has a language-like property, viz. the relationship between the mental model and that which it is a model of. But the similarity is striking only if one assumes that meaning is representational in nature. If we assume that it is instructional in nature, involving the act of drawing on conceptual resources rather than the interpretations produced by drawing on those resources, the division of labour between the model-building capacity and human language becomes more natural. Meanings are instructions, on the basis of which we perform certain 'cognitive actions' – and among those actions are the building of mental models. Other cognitive actions could be: referring from words to entities of reality or rather to perceptual representations of reality, restructuring of models, comparison of models, and so on.

Meaning is a meta-phenomenon in relation to the representational level, so to speak. Human language does not 'represent' or 'denote' models or those things of which they are models – it serves as a medium prodding interlocutors to produce and manipulate the models. And, more generally, prodding interlocutors to produce the understanding on which communication depends.

This means that lexical items are not strictly speaking representations of anything in the mind or the world. The word 'cow' does not 'stand for' the monolithic concept 'cow' (as constituting the putative integrated level above the connections). Rather, the word 'cow' is an instruction to draw on the relevant mental resources, embodied in the net of associations, to produce a mental model or representation appropriate to that word, as required by the current purposes of the interaction. What we come up with depends on what context we have: whether we are thinking of the bull's mate, the source of milk, beef, the price of brides, or the calf's mother; but there is no reason to think that there is a different meaning potential for 'cow' in each case.

This view of meaning, as part of a picture where understanding is 'intent-driven', means that the mechanism that creates metaphors is part of that process of understanding which is required to make sense out of all utterances. The audience always has to get the best possible sense out of the instructions that are linguistically encoded in the message. New metaphors arise when there is a clash between instructions; but even when there is no such clash, there is always an element of finding out precisely in what way the existing potential makes sense in this particular context, and the type of conflict that we associate with metaphor is only a special case of this more general phenomenon.

Such cases also illustrate the constant interplay between meaning potential and actualization. It is necessary to have a meaning potential to start with, or the process of interpretation would not have anything to go by; but when a new combination of elements is used, the process of interpretation may necessitate the assignment of a type of actual meaning to an element that was not part of the potential meaning that existed before. To take a case that as far as we know is new, let us imagine that someone uses the phrase "he is a football elephant". The audience would then invoke the 'elephant' net in trying to determine what sort of entity this phrase is meant to convey, and invoke it in such a way that the 'football' net of associations could be used to characterize the entity selected. If it turned out that the phrase meant (= was intended to be interpreted as designating) a football-player who was as massive and inflexible as an elephant, the two nets of associations would have interlocked in a new way, setting up a link between the potentials of the two words – and the interlocutors might next time only have to talk about 'elephants' without the modifying 'football', so that the meaning potential of 'elephant' had expanded marginally as a result.

The difference between interpretation of normal syntactic combinations of words (i.e. when "a football elephant" means 'an elephant playing football in circus') and metaphorical interpretation of new combinations lies primarily in the fact that the new combination changes the potential meaning of the words and normal combinations do not. The potential determines the process of establishing the actual meaning; but the actualized meaning in turn influences the future meaning potential. The death of metaphors is a special case of semantic change through use.

The reason why the distinction between literal and non-literal meaning is generally felt to be necessary is that it points to an obvious truth, viz. that in describing the meaning of a word or phrase in context we can distinguish between more constant and basic aspects of its meaning and more marginal and variable aspects. But the concept of literal meaning gives a simplistic account of this polarity. To understand what goes on we need two distinctions. The first is the distinction described above between potential and actual meaning. The second is a distinction between central and peripheral aspects of the meaning potential. As described above in the case of 'cow', the natural kind 'cow' can be assumed to form the center of the net of associations involved in the word – if we eliminated that aspect of the net, it is difficult to imagine what would be left. Other aspects of the meaning potential are more dispensable; one could remove the culturally central role of cow milk as opposed to e.g. pig milk and leave most of the meaning of the word 'cow' intact. However, some uses of the word cow would be ruled out, as in 'I am not a very good cow!', a summing-up by an acquaintance of ours when she found that all her children refused to be breast-fed as soon as they realized that there were alternatives.

Those aspects of meaning that are put forward as plausible candidates for literal meaning are generally those that are historically prior, and which have formed the necessary basis for other shades of meaning, so that the extended meanings would not have been possible without this 'basic' meaning. To the extent that there is a part of the meaning potential of a word that is presupposed by other parts of the meaning potential, there are good reasons for giving it special status; in case of clearly defined core areas with a corona of gradually more marginal cases, we get the classical prototypes (cf. Rosch et al., 1976). In other cases we get less clearly defined centres, as in 'game', Wittgenstein's classical example; there may also be bi- or tri-focal cases, where more than one central area can be contrasted with other, more marginal areas.

The two distinctions preserve what is worth preserving in the distinction between literal and non-literal meaning without committing us to any untenable idealizations. By insisting on the distinction between potential and actual meaning, with potential meaning being understood as constituted by the whole net of associations, we emphasize that in all actual uses of a word there is an element of interpretation – no matter how close to the 'standard' or

'basic' meaning we may be. By assuming degrees of centrality as a feature that may be built into the structure of the meaning potential, we can avoid the chaotic situation that the notion of literalness is felt to protect us against, without taking on the task of dividing all meanings into two neat areas, literal and non-literal (cf. also Bartsch, 1984).

9. An integrated model of understanding

Summing up the claims that we have made so far, we can say that understanding involves recognition of types and also the ability to recognize elements as identical across contexts, but both types of skills must be seen as subordinate to the fundamental task of recognition of intentions. It now remains to be a bit more precise about the division of labour and interaction between the components involved in this overall picture, and to see how the view of linguistic meaning argued above fits into the picture of understanding.

'Softness', as pointed out by Smolensky, is ubiquitous in the sense that it cannot be put aside as a low-level phenomenon superseded at higher levels by 'hard' categories. Not only at the level of individual words, but at the levels of all conceivable combinations, recognition processes involving vague elements of association may influence interpretation in ways that cannot be captured by algorithmic (i.e. purely compositional) rules. All 'chunks' are susceptible to processes of that kind, up to and including whole utterances. However, such processes do not have the last word. If they did, we would be back with stimulus and response as the bottom line in accounting for understanding, however refined the connectionist versions are compared with e.g. Bloomfield's.

Further, human interpretation of linguistic texts is not a matter of calculating the content as something that is 'in' the text – it involves intentions which are not part of the text as an object in itself. In principle any human interpretation process is the mediation of intentions and external linguistic stimuli. Any stimulus is interpreted under a certain aspectual shape (cf. Searle, 1990) which is not emergent from the stimulus itself but depends on the intentional stance of the human being.

One way in which the PDP-type processes are constrained is via the compositional element in understanding, i.e. those aspects that are due to the meanings of constituent elements. For each linguistic element, there is a meaning potential which is invoked, and whose integrity must be respected in working out the total meaning. When the word 'coffee' is used in an utterance, the ultimate understanding must contain an element of locating an actualized meaning for the word 'coffee' based on the (socially recognized) potential meaning of the word.

What we see, then, is partly that compositional meanings are (part of) the input at all levels to processes of type recognition. If component elements dictated the types that were possible, 'hardness' would have the last word. If understanding of linguistic utterances could freely ignore individual components in favour of an overall assessment, 'softness' would have the last word. In fact neither is the case. Typification must respect the integrity of individual components, but understanding has to go beyond any computable aggregate of components. The actual meaning (=interpretation) of a text is always underdetermined by the component meanings.

The basic reason for the insufficiency of compositional understanding is expressed by the notion of intent-driven understanding, which is also the second way in which the type-recognition processes are constrained in human understanding. As pointed out above, understanding an utterance is basically to understand what it does in context, which means what it does as a token rather than what it is in terms of a type. This means that type-assignment on whatever level cannot be final, until an interpretation has been found that is satisfactory in terms of the 'recognition of intention' aspect.

At the level of whole utterances, certain 'speech act' properties are important 'typification' aspects. If there is a speech act type of 'promise' in the speech community, interlocutors need to recognize something as a promise when that is appropriate (whatever additional properties the utterance may have). But the naked output category of 'promise' is not the sum total of understanding. Understanding involves the whole process of working out what the utterance means in the actual position of the interlocutors – in other words seeing the utterances as bringing about a pragmatic change in the here-and-now situation of the human agents concerned, rather than merely as instantiating a type.

Two examples may serve to illustrate this picture. *Can you pass the salt?* is immediately recognizable as a polite request. But according to the constituent elements it could also mean 'are you able to go past the salt (desert)?' In understanding an utterance involving these words, we disambiguate 'pass' as meaning either 'hand over' or 'go past', we understand 'salt' as referring either to a salt caster or a salt desert, and possibly we understand 'you' as either the addressee or the 'generic' you. It is unlikely that we would ever understand it the second way; but the option is open and would come into its own if the question of crossing a salt desert on foot arose, and somebody spoke such an utterance standing outside in the blinding sunlight, staring into the shimmering distance. Neither type recognition nor compositional computation can take us all the way; only recognition of intention could produce anything worthy of the name of understanding.

The utterance *the policeman attacked the black man with the knife* is, linguistically speaking, ambiguous with respect to who carries the knife. The audience is free to decide where the knife is, based on preconceptions involving recognizable situation types. They are also free to form a picture in their mental model in response to the phrase *the black man* in which he comes out as a dope-peddling dropout – again, linguistic meaning underdetermines understanding. In *with the knife, the policeman attacked the black man*, the situation is different. It is still possible to imagine a dope-peddling dropout if one is so disposed; but with respect to the knife, the utterance is not ambiguous.

In many cases, empirical processes of understanding would probably still put the knife with the black man (cf. Miller, 1951, ch. 12: 'The Social Approach') – because the situation where policemen attack people with knives does not belong to the pre-existing set of categories. However, it is part of playing the game of language that the compositional rules are valid in spite of what an overall assessment purely based on the 'soft' approach would lead to. This is true even if the speaker did in fact want to place the knife with the black man, but got mixed up because he also wanted to front the phrase 'with the knife'. In that case we would want to say that there was a difference between what he said and what he wanted to say – which would be impossible if there were no 'hard' rules of language, and understanding was probabilistic all the way through.

The notion of intent-driven cognition also involves a basic difference in relation to the view of linguistic meaning as conceived both in AI and PDP. Both camps adhere to a notion of meaning that is ultimately causal in nature. Fodor (1987) discusses how one can set up an account of meaning that avoids all the familiar fallacies of the crude causal theories of meaning, while remaining ultimately causal in nature. In PDP, as pointed out above, processing takes place at a sub-symbolic level, replacing the notion of a sign or symbol with input–output relations. What takes the place of meaning is the process of excitation of neurons by the input. This process is clearly causal in nature, and close to some of the cruder causal theories that Fodor wants to avoid.

In the terms used above, both views of meaning are backward-oriented, each on their level – AI on the level of constituent symbols, PDP on the level of output categories. This is logical if we want machine processes to have the last word in the process. If assigning meaning to an utterance involves basically that you have to find out what the utterance is designed to do for you in a concrete situation, we put the machine out of business as a meaning-assigning device – which is in fact what we have to do. Machines have no intentions, and do not recognize them in human beings; if they did not work in a fully deterministic manner, programmers would have a rather more difficult time.

10. Machines, intentions and the Turing test

Before we take the last step of the argument, there is a familiar problem that needs to be placed in the context of the issue under consideration. The

11. The complementarity of AI and PDP

We have now tried to demonstrate at some length why AI and PDP cannot give a complete account of cognition, either alone or in combination. It remains to look at what they can do in the perspective outlined.

What takes the place of intentions and purposes in computer processing is the task task set by the programmer. In the language of computation: the grammar does not compute anything without a parser and a 'go' command. The 'go' command is the instantiation of an intention in the machine. It means something like: use the rules of grammar until you have generated a representation (a tree structure) of the input string with the top node 'S'.

The machine is, so to speak, commanded to have the intention to recognize the input as an instance of a grammatical sentence. The human addressee, on the other hand, has the intention to recognize the input as an instance of a social action which makes sense. The computational procedures, whether based on compositionality or on pattern recognition, are unable to carry out the full disambiguation process as performed through human cognition, because it ultimately involves recognition of intention, as argued above; it only 'has the intention' to do precisely what it is told, which in the case in question is to recognize the input string as generated by the rules of the grammar.

What this means is that the programmer must specify tasks to be performed to such a point that the task is sealed off from interference by intentional mechanisms, before the computer can perform them. This is why issues involving 'modularity', 'autonomy' and 'impenetrability' of subtasks of the organism is so important in defining the scope of computer simulation. But we need not assume built-in autonomy in order for computer simulation to be useful – we only have to assume that there are subtasks that are sometimes performed autonomously, and then see the goal of simulation as capturing ways in which cognition can solve such subtasks.

In that perspective, AI-like processes and PDP-like processes may be seen as complementary, not only in language understanding, but generally. We have seen how fitting utterances as a whole as well as their component parts into pre-existing types was an important part of understanding. We have also seen how conflicts sometimes had to be resolved in favour of compositional rules, rather than gradual overall assessment. It has sometimes been claimed that one type of ability was inferior to the other. Adherents of AI have claimed that PDP-like skills were 'primitive' compared with the precise logic of AI procedures; and opponents of AI have claimed that expert performance was being able to do without rules (cf. Dreyfus and Dreyfus, 1986, as quoted by Smolensky, 1987: 138). But it may be better to see the processes as being good for different purposes.

To exemplify this, we may take the case of a bird-watcher. It is easy to see

that expertise in bird-watching has some PDP-like properties. The list of bird species of the world constitutes a pre-existing set of output categories, and an expert can assign a large proportion of the specimens he comes accross to one of these species with great accuracy 'at a glance'. Where the beginner has to look at the colour of the primaries and the length of the tail etc. and often gets it wrong, the expert does not need to worry about individual features and almost always gets it right. To say that experts had to go through precise logical steps each time they made a judgment would obviously be implausible; and often a beginner will worry about a white spot that should not have been there, which the expert has hardly noticed because he can confidently say that this must be an albinistic specimen: confidence in overall assessment permits negligence towards details. Therefore, experts need more than rules.

But if the expert was like a PDP processor, he would have no possibility but to make overall assessments in relation to pre-defined output categories, mechanically putting everything in places he knows beforehand. A human expert does not limit to that. He may, for instance, first categorize a specimen according to his pre-established categories, and then start doing exactly what the rankest beginner would also be told to do: note down carefully each individual feature of the bird, going through it in a step-by-step fashion. On his return he may then compare with museum specimens and discover that this was the subspecies from Siberia, with which he was not familiar. Analogously, the human addressee may come up with an interpretation based on overall assessment, and then have misgivings, think carefully about a component element of the utterance and revise his interpretation.

True competence, therefore, involves knowing when to use what type of processing and being able to change gears between them. In the controversy between AI and PDP this means that we need both to be able to test phenomena that we work on against our preconceived output categories (as in PDP) and see what that yields – and to be able to work with some pre-defined input elements and see where we get when we put them together (as in AI). The hermeneutic circle, where parts and whole mutually presuppose each other, strictly speaking means that understanding is never achieved; and there is an important truth in that. Understanding, as part of the process of life, always involves moving forward from what we know to something new that needs to be integrated into our position, which changes in the process. If cognition was not open-ended it would be a bad servant in carrying on this process.

References

Bartsch, Renate, 1984. The structure of word meanings: Polysemy, metaphor, metonymy. In: Fred Landman and Frank Veltman, eds., Varieties of formal semantics. Dordrecht: Foris.

Bharucha, Jamshed J. and Peter M. Todd, 1989. Modeling the perception of tonal structure with neural nets. Computer Music Journal 13(4): 44–52.

Fodor, Jerry A., 1987. Psychosemantics: The problem of meaning in the philosophy of mind. Cambridge, MA: MIT Press.

Fodor, Jerry A. and Zenon Pylyshyn. 1988. Connectionism and cognitive architecture: A critical analysis. Cognition 28: 2–71.

Grice, H.P., 1957. Meaning. Philosophical Review 66: 377–388.

Johnson-Laird, Philip N., 1977. Procedural semantics. Cognition 5: 189–214.

Johnson-Laird, Philip N., 1983. Mental models. Cambridge: Cambridge University Press.

Johnson-Laird, Philip N., 1988. The computer and the mind. London: Fontana.

Kamp, Hans, 1981. A theory of truth and semantic representation. In: Jeroen Groenendijk, Theo M.V. Janssen and Martin Stokhof, eds., Formal methods in the study of language (= MC-tracts 135), 277–322. Amsterdam: Mathematical Center.

Lakoff, George, 1987. Women, fire, and dangerous things: What categories reveal about the mind. Chicago, IL: Chicago University Press.

Langacker, Ronald W., 1987. Foundations of cognitive grammar, Vol. 1: Theoretical prerequisites. Stanford, CA: Stanford University Press.

Langacker, Ronald W., 1991. Foundations of cognitive grammar, Vol. 2: Descriptive application. Stanford, CA: Stanford University Press.

Lewis, David, 1972. General semantics. In: Donald Davidson and Gilbert Harman, eds., Semantics of natural language, 169–218. Dordrecht: Reidel.

Loy, D. Gareth, 1989a. Preface to the special issue on parallel distributed processing and neural networks. Computer Music Journal 13(3): 24–27.

Loy, D. Gareth, 1989b. Preface to the second special issue on parallel distributed processing and neural networks. Computer Music Journal 13(4): 10–11.

McGinn, Colin, 1989. Mental content. Oxford: Blackwell.

Millikan, Ruth, 1984. Language, thought, and other biological categories. Cambridge, MA: MIT Press.

Miller, George A., 1951. Language and communication. New York: McGraw-Hill.

Scarborough, Don L., Ben O. Miller and Jacqueline A. Jones, 1989. Connectionist models for tonal analysis. Computer Music Journal 13(3): 49–54.

Penrose, Roger, 1989. The emperor's new mind. London: Oxford University Press.

Rosch, Eleanor, C.B. Mervis, W.D. Gray, D.M. Johnson and P. Boyes-Braem, 1976. Basic objects in natural categories. Cognitive Psychology 8: 382–439.

Schank, Roger C. and Robert P. Abelson, 1977. Scripts, plans, goals and understanding. Hillsdale, NJ: Erlbaum.

Searle, John R., 1980. Minds, brains and programs. The Behavioral and Brain Sciences 3: 417–457.

Searle, John R., 1983. Intentionality. Cambridge: Cambridge University Press.

Searle, John R., 1984. Minds, brains and science. London: BBC. [Reissued 1989, Harmondsworth: Penguin.]

Searle, John R., 1990. Lectures on language and mind at the Nordic Symposium, University of Copenhagen.

Smolensky, Paul, 1987. The constituent structure of connectionist mental states: A reply to Fodor and Pylyshyn. The Southern Journal of Philosophy 26, Supplement, 137–160.

Sperber, Dan and Deidre Wilson, 1986. Relevance. Communication and cognition. Oxford: Blackwell.

Winograd, Terry, 1976. Toward a procedural understanding of Semantics. Revue internationale de philosophie 3: 117–118 and 260–303.

INSTRUCTIONS TO AUTHORS

The *Journal* publishes the following types of contributions: (1) *full-length articles* on all topics of linguistic pragmatic interest, (2) short, free-form *squibs* intended to clarify theoretical standpoints, try out new ideas, report research findings, or simply to entertain or stimulate discussion among colleagues, (3) *replies and rejoinders* to previously published articles, (4) *reviews* of books not more than five years old, (5) *review articles* on particularly noteworthy or thought-provoking new books, and (6) brief *book notices*. Three types of issues are published: *Regular issues* containing articles on different aspects of pragmatic theory and research. *Special Issues* on topics of particular relevance to current discussions in linguistic pragmatics, and *Focus-on-Issues* on broad subject areas of interest to different groups of readers. Manuscripts accepted for publication are automatically considered for inclusion in each type of issue.

Manuscripts. Manuscripts should be sent to the Chief Editor, Jacob Mey, Odense University, Campusvej 55, DK-5230 Odense M, Denmark, and should be prepared in accordance with the guidelines summarized below. *Manuscripts not conforming with these guidelines will be returned to the authors for revision before being further processed. Also, the Journal does not support duplicate submissions: only manuscripts offered exclusively to the Journal can be considered for publication.* The preferred language of publication is English (although exceptions are occasionally made), and manuscripts should meet the *Journal's* English language standards (see below). Authors are expected to provide *four* copies of their contribution, typed *triple-spaced*, with wide margins, on plain, standard A4 or U.S. letter size paper. Papers should be typed on one side of the page only, and the pages should be numbered consecutively. The following information should appear on the first page: the full title of the contribution, the author's name and full address, and an abbreviated title not exceeding 40 characters including spaces. An abstract not exceeding 200 words should be provided, as well as a short biographical note not exceeding 100 words per author. Diskettes are welcome only when accompanied by hard copy manuscripts.

Citations. Quotations should be enclosed in double quotation marks. Single ('scare') quotes may be used to draw attention to a particular item in the text. Words from other languages, and words intended to be especially emphasized, should be underlined (italicized). References in the text to publications should include the author's name, the year of publication, and, if necessary, page numbers, as in the following example:

Although this type of conversational coordination has been investigated from different points of view over the years (Sacks et al. 1974, Van Dijk and Kintsch 1983, Schiffrin 1987a,b), a general theory of language use accounting for it, is still, in Kasher's (1991a: 129) words, "..."

Bibliographical References. A reference list containing all works referred to in the text and footnotes, alphabetically arranged and typed *triple-spaced*, should be included at the end of the manuscript. Authors' *full* first names should be used unless the authors themselves customarily use only initials. References to two or more works by the same author in a single year should be accompanied by a lower-case a, b, etc. after the year of publication, both in the reference list and in citations in the text. References to books should include the place of publication and the publisher's name, and references to articles in journals should include volume and page numbers, as in the following examples:

Kasher, Asa, 1991a. Pragmatics and Chomsky's research program. In: A. Kasher, ed., The Chomskyan turn, 122-149. Oxford: Blackwell.
Kasher, Asa, 1991b. On the pragmatic modules: A lecture. Journal of Pragmatics 16: 381–397.
Sacks, Harvey, Emanuel A. Schegloff and Gail Jefferson, 1974. A simplest systematics for the organization of turn-taking for conversation. Language 50: 696–735.
Schiffrin, Deborah, 1987a. Discourse markers. Cambridge: Cambridge University Press.
Schiffrin, Deborah, 1987b. Toward an empirical base in pragmatics. Language in Society 16: 381-396.
Van Dijk, Teun A. and Walter Kintsch, 1983. Strategies of discourse comprehension. New York: Academic Press.

Footnotes. Footnotes should be typed *triple-spaced* on a separate page and numbered consecutively. They should be as few and as short as possible, and should not contain bibliographical references except insofar as these are relevant to the matters discussed in the footnotes themselves. All other references should be given at the appropriate places in the main text.

Figures. All tables, lists, figures, illustrations, and other graphics must be reproducible originals, and should be presented with appropriate captions on separate sheets, with instructions indicating approximately where they are to appear in the text. Phrase structure trees, argumentation schemata, networks, flowcharts, and diagrams should be kept to a minimum. Figures should be freely movable in the text, and should be referred to by number (e.g., "figure 2"), and *not* by expressions such as "the figure below", or "the figure above". Illustrations should be drawn in India ink in large, reproducible size, and should be carefully lettered. *Authors will be charged if their illustrations have to be re-drawn.*

Symbols. Current type-faces should be used, and special symbols should be avoided as much as possible. Most standard logical symbols are printable, as well as the Greek, Cyrillic, Gothic, and phonetic alphabets, Chinese characters, and Japanese *kanji/kana*. All other symbols from scripts other than those above, must be drawn, in oversize format, in India ink, on separate sheets, following the procedure outlined for illustrations. *In cases of doubts or questions, authors are advised to consult with the Editors before submitting their manuscripts.*

Style. The Editors reserve the right to correct, or to have corrected, non-native use of English (or other languages). The *Journal* has adopted U.S. English usage as its norm (this norm does not apply to other native users of English). Also, the *Journal* is actively committed to avoiding sexist language as far as possible, and to affirming and promoting the use of nonsexist expressions. If in doubt, authors should consult the *Guidelines for Nonsexist Use of Language* of the American Psychological Association, published in the *American Psychologist* 30 (1975): 682–684, or the *Guidelines* of the Linguistic Society of America in the *LSA Bulletin* (# 135, March 1992).

Authors' benefits. Upon publication in the *Journal of Pragmatics*, authors receive (1) 50 reprints of their contribution free of charge, and (2) a 30% discount on all Elsevier Science Publishers books.

US mailing notice – **Journal of Pragmatics** (0378-2166) is published monthly by Elsevier Science Publishers (Molenwerf 1, Postbus 211, 1000 AE Amsterdam). Annual subscription price in the USA US $350.00 (subject to change), including air speed delivery. Application to mail at second class postage rate is pending at Jamaica, NY 11431. USA POSTMASTERS: Send address changes to Journal of Pragmatics, Publication Expediting, Inc., 200 Meacham Avenue, Elmont, NY 11003. Air freight and mailing in the USA by Publication Expediting.