Semantics, pragmatics, and the digital information age

Autor(en): **Dresner, Eli / Dascal, Marcelo**

Objekttyp: **Article**

Zeitschrift: **Studies in Communication Sciences : journal of the Swiss Association of Communication and Media Research**

Band (Jahr): **1 (2001)**

Heft 2

PDF erstellt am: **15.08.2024**

Persistenter Link: https://doi.org/10.5169/seals-791148

Studies in Communication Sciences 1 (2001) 1-22

Eli Dresner & Marcelo Dascal*

SEMANTICS, PRAGMATICS, AND THE DIGITAL INFORMATION AGE

In the first section of this paper we tell a well known story. The computer is a machine that performs tasks that are traditionally viewed as *cognitive*, e.g. numeric computation. It was therefore hoped that we shall quickly teach this machine complex cognitive abilities, many of which are related to *natural language processing*. However, these high hopes proved to be premature: our understanding of linguistic meaning (semantics) and linguistic use (pragmatics) is still too meager for us to simulate complex language processing computationally.

In the second section we present the current, less acknowledged part of this story, having to do with the *digital information age*. At first blush it seems that by moving to this age we leave the problems described above behind us: the information age is built around computer technology that helps us communicate better with each other, not technology that is supposed to simulate our thought processes and our speech; therefore it looks as if the struggle with semantics and pragmatics is no longer required. It turns out, however, that things are different: we show how (and why) semantics and pragmatics prove to be central to computer technology in the information age as well.

Keywords: Semantics, Pragmatics, Information Age, Artificial Intelligence.

* Department of Philosophy, Tel Aviv University (Israel), dresner@post.tau.ac.il and dascal@spinoza.tau.ac.il.

1. Computers and Natural Language

1.1 Computation and Language

Computers have been closely and deeply connected to language and linguistic processing much before the term Artificial Intelligence was first coined, and thus also before any attempts were ever made to write artificial intelligence programs that tackle natural language. In fact, as we now see, computers were inherently connected to language even before there were any computers.

The computer was first conceived as a machine that performs computation. When this machine was first built, its creators were not required to begin their work by articulating clearly and formally what computation is; they could rely on a body of theoretical work put together by mathematical logicians in the first part of our century. At the heart of this body of work were several exact definitions of what computation consists in; these definitions were proven to be essentially equivalent, and therefore it was famously conjectured (by A. Church, in what is called Church's Thesis) that each of these definitions captures fully our intuitive notion of computation (Shoenfield 1967: 119).

The seemingly most direct way to define computation is through talk of numbers: in everyday speech to compute is to perform certain numerical manipulations. However, not every procedure involving numbers deserves to be called a computation; for example, picking up from any pair of numbers the one that seems to you prettier is not (and should not be defined as) a computation. Addition and multiplication, on the other hand, are computations; similarly, choosing the larger of two numbers is a computation. It is a not a trivial problem to generalize these intuitions and say exactly what operations (what functions) on numbers are computations. The solution to this problem is called *the class of recursive functions* — a class of procedures that is clearly defined and that matches our intuitions as regards what a computation is; however, this solution turns out to be much less direct and intuitively accessible than could be first expected.

Surprisingly enough, the other main ways of defining computation involve in an essential way not numbers, but *languages*. To be exact, these definitions involve formal languages — sets of strings of symbols taken from some finite alphabet. One of these ways relies on the notion of a *formal grammar* — a set of rules according to which strings can be transformed until they reach some final state. A computation can be defined as a procedure that can be carried out by such a set of rules (Lewis and Papadimitriou 1998: 224-232). This way of getting at computation was utilized by Chomsky in his work on natural languages, and we shall return to it presently.

The other, more well known and influential way of defining computation via (formalized) language is through the abstract notion of a Turing machine, named after its originator, A. Turing (Boolos and Jeffrey 1974: 19-33). At any stage along its operation such a (hypothetical) machine can read a symbol off a (similarly hypothetical) tape, and then change the symbol, erase it, or move left or right along the tape — according to the input symbol the machine has just read and its internal state. Such simple, abstract machines can carry out all (and only) computations; that is, a computation can be defined as what a Turing machine can do. In a sense, all existing computers are electronic realizations of Turing machines, and thus they can be rightly described as devices that operate on strings of symbols, i.e. that deal with languages.

We see, then, that from its inception the computer had close ties with languages, albeit formalized, 'syntacticalized' languages.

1.2 Why Tackle Natural Languages Computationally?

It could be asked now why would anyone expect computers to deal with *natural* languages. We do not expect other pieces of equipment (such as refrigerators and car engines) to evolve into things that can speak, so why should we have different expectations from a machine that calculates?

The answer to this question is implicit in what was said in the preceding subsection. As opposed to the refrigerator and the engine, the computer was developed on the basis of a theory that tried to capture a cognitive ability of ours — the ability to compute. The computer performs computation, which is a mental task (again, as opposed to refrigeration and the production of motion); therefore it makes perfectly good sense to ask what cognitive tasks can be carried out (or modeled) by the computer, i.e., what things that we do in our mind can be reduced (in some sense) to computation, to symbolic manipulation. This question (and the various answers to it) can be justifiably said to be at the center of the development of the computer, from its first construction in the late forties onwards.

What is most special about our minds is that we can *think*: we can reason and conjecture, deduce and induce, make hypotheses and rule them out, and so on. These abilities enable us to understand the world around us and control it according to our needs. Can these tasks be performed (or simulated) by the computer? Well, before this question can be answered a major point has to be acknowledged and faced. These higher cognitive abilities are intimately related to our ability to use *natural language*: we reason in language, we make inductions and deductions in linguistic terms, and we think about the world around us using language. (Some authors go as far as to claim that high-level cognition is coached in natural language [Dascal 1999].) Therefore, in order for the computer to be able to perform these tasks it has first to be taught to deal with natural language, or with a symbolic system as expressive and powerful as natural language. (Some cognitive scientists hold that we think not in natural language, but rather in a special language of thought; for example, see Fodor 1994. This point need not bother us here, though: the task of teaching the computer the language of thought [Mentalese, in Fodor's terms] would be as formidable as teaching it any natural language.) This is one avenue leading to an attempt to reproduce language use on the computer.

Another, perhaps more direct avenue that leads in the same direction is this. One of our main higher cognitive abilities is manifested in linguistic communication: people speak to us, and we *understand* what they say. Indeed, we accumulate much of our knowledge of the world around us through listening to speech and reading, i.e., via language, rather than through direct experience. For this reason too language processing would be a natural target for a machine that is supposed to reproduce (or simulate) mental processes.

The last two paragraphs give expression to the following important point: language is central both to *communication* and to *thought*. For this reason the computer is connected to language in two ways — one through thinking, and the other through the understanding of speech. Both capacities (thought and speech processing) are mental, both involve language, and therefore both capacities require the mastery of language by the computer if they are to be reproduced by it. We shall return to this point in various forms below.

1.3. Syntax, Semantics, Pragmatics

Now what is it that we need to teach the computer about natural language? What does our mastery of language consist in? This mastery is usually broken up into three components — syntax, semantics and pragmatics (Morris 1938).

Human *phonological* competence is the capacity to translate auditory input into the syntactic building blocks of language, i.e., words, and, on the other hand, to produce the phonetic representation of linguistic expressions. The first component, our *syntactic* knowledge, consists in our ability to recognize well formed expressions in our language, to segment such expressions according to their grammatical structure, and to produce grammatically correct expressions. These abilities are traditionally characterized as not involving considerations of meaning: a judgment whether a certain string of symbols (communicated to us auditorily or visually) is a legal expression in our language can be made independently of the question what this expression means (if indeed it is well formed).

The second and third of the components listed above have to do with the meaning of linguistic expressions, not merely with their form. Our *semantic* knowledge is our knowledge of what the expressions in our language mean, plain and simple: what is signified by various words, and what is the meaning of the complex expressions that can be made from these words. In the formation of complex linguistic expressions from more simple ones syntax comes into play, so our semantic knowledge relies on our syntactical knowledge; however, our semantic abilities clearly go beyond syntax.

Finally, we get to *pragmatics*. Our pragmatic knowledge too has to do with meaning; it consists in our ability to use meaningful linguistic expressions in the many ways that we do. In order to see better what these ways of use consist in, note how many things we might be doing when we use language, e.g., in making an utterance. We might simply want to inform someone of something, of course, but we might also want to convey an idea metaphorically, to imply a conclusion, to bring an issue into consideration, to frighten, to promise, to warn — and many, many, more things. All these things take some knowledge to do — knowledge that relies on our knowing the meaning of linguistic expressions, but that goes beyond it. This further kind of knowledge of language is called pragmatic.

Now recall that we ended the previous subsection by noting that language is used as a vehicle both of thought and of communication. It follows that our pragmatic capacities apply to these two domains, the external and the internal, and hence we must distinguish between two types of abilities: our Sociopragmatic abilities, having to do with language use in communication, and the Psychopragmatic ones, having to do with how language is used in thought. The respective domains in language research are Sociopragmatics and Psychopragmatics (Dascal 1992: 152-153; and the relevant bibliography therein). Consider for example the notion of relevance, which is central in contemporary pragmatics (cf. Grice 1975; Dascal 1977; Sperber and Wilson 1986). Sociopragmatics is concerned with the important role this notion plays in communication, while Psychopragmatcs investigates the equally important (albeit different) role played by relevance relations in cognitive processes. No doubt these two domains are inter-related (e.g., the apparent communicative irrelevance of an utterance in a conversation triggers an abductive search for an interpretation of the utterance that preserves its presumed relevance), and Relevance Theory has emphasized the cognitive underpinning of communicative relevance. Nevertheless, it is important to distinguish the two domains for a variety of reasons — some of them directly connected to the theme of this paper (See section 1.4 below).

1.4. Successes and Failures

So how successful have we been in reproducing on the computer the capacities and abilities described above? In a nutshell, the answer is arguably this: the more we have made progress in understanding any of the human abilities in question, the more successful we have been in simulating these abilities on the machine. Let us elaborate and explain this claim, going through the three categories of linguistic competence listed in the previous subsection.

a. First and foremost comes syntax, not only because it appears first on the previously given list, but also (and mainly) because (a) linguists have first developed a reasonable understanding of the syntax (rather than semantics) of natural languages, and because (b) the computational treatment of natural language syntax has been first to evolve and also first in its accomplishments so far. Indeed, reason (b) should come as no surprise: we saw in subsection 1.1 that several of the formal representations of the notion of computation take the form of syntactic manipulations on strings of symbols — expressions of formalized languages. Thus it is not a great leap to hope that the syntactical manipulations that we perform in processing natural language will be close enough to those undergone by formal languages in computation — thereby making natural language syntax computational and therefore amenable to reproduction on the computer.

The Chomskian turn in linguistics consists in the fulfillment of this hope, at least to a large degree. Chomsky and his followers have gone a long way in showing that at its core the syntax of all natural languages can be produced by a rather simple kind of computational mechanisms (called Context Free Grammars), enhanced by a plethora of further devices (e.g., transformations) that are computational in character as well (Chomsky 1965). These mechanisms and devices are theoretical entities, like Turing machines; however, according to Chomsky they are realized in our brain, as Turing machines are realized by digital computers.

b. This initial success with syntax seemed in the sixties and early seventies to pave the way for similar successes in semantics. (The existence of pragmatics as a distinct body of knowledge [and as an area of research] was not widely acknowledged at the time.) However, before one can begin to aspire to such a success there is an important point that must be faced. It is quite clear what it is for a computer to reproduce our syntactical abilities, because its own abilities are syntactical in nature. Thus a program that computes the syntax of English would count as adequate if its output on English strings (i.e., English expressions) — e.g., evaluations of grammatical correctness — would be similar to ours. With semantics, on the other hand, things seem to be completely different: we are far from being able to say precisely what it is that we know when we know meaning, and are not any nearer to being able to say how (if at all) knowledge of meaning can be reproduced on a computer. Indeed, there are those (like J. Searle) who hold that meaning and computers are inherently divorced: according to Searle's view (Searle 1984) a merely computational process that is executed by a computer does not (and cannot) have any semantic aspect to it.

There is a way to get around this problem, at least in part. Our semantic capacities have *symbolic manifestations*; that is, the semantic processing that we perform can be described as beginning and ending with English expressions, construed as mere syntactical entities. Therefore a purely syntactic process within the computer could be said to represent (in some sense) our own semantic capacities if its output on some input would be similar to ours (on the same input). This way we can put aside the deep philosophical and scientific questions about meaning and thought, and still deal (albeit indirectly) with semantics. Let us give two paramount examples of this strategy.

The first example is that of *inference*. For any two sentences it could be asked whether the second can be inferred from the first. (More generally, for any set of sentences A and a further sentence S it can be asked if S can be inferred when the sentences in A are taken as premises). The answer to this question certainly has to do with the meaning of the sentences in question, not with their grammar alone; thus inference is a semantic notion. However, the two sentences can be looked at also merely as strings of symbols that could be the input and output of a computer program, and therefore it can be coherently asked if the computer can mimic our inference processes, i.e., whether for any pair of English sentences the computer would say that the second follows from the first exactly when we would say so too. Indeed, for some relatively simple formalized languages the answer to this question has proven to be positive: for such languages there are purely syntactical procedures that are in complete accord with our semantic intuitions as regards inference relations among sentences.

The second example is *translation*. For any sentence from a given language it can be asked what its correct translation into another language is. As in the previous example, this question is semantic — it has to do with what the sentence from the first language means — but it too has a purely syntactic counterpart: it can be asked whether a computer program can be written that mimics translation, i.e., that given an input string from the first language produces an output string from the second language that is a correct translation of the input — correct according to our semantic intuitions.

Having pointed out concrete, coherent questions about our ability to reproduce (albeit indirectly) semantic knowledge on the computer, we can turn to consider their answers. Roughly put, the answer to both is this: «We have made some progress (in the computerized reproduction of inference and translation), but not as much as we had hoped for; we have not been successful yet, and are much less sure than before that success is forthcoming.» The reason for this state of affairs is that the logical and semantic analysis of natural language that is required for the two tasks described above has proven to be much more difficult than first thought. For example, many *logical constructions* in natural language have been found to be beyond the scope of well known and well behaved formal logical mechanisms (Barwise and Cooper 1981); also, *vagueness* is a natural language phenomenon that is very hard to deal with (Kamp 1981, van Deemter and Peters 1996). Progress on these fronts (and others) has been made, but it is not as rapid as was hoped for thirty years ago. Furthermore, the analysis of the semantic structure of the lexicon has proven to be a formidable task (Jackendoff 1983, 1987; Pustejovsky 1993). As these and other authors have argued, lexical semantics is inherently tied to a perceptual and conceptual structures, and therefore a systematic (and, hopefully, computational) description of the lexicon can be provided only within the scope of a broad theory of cognitive psychology.

c. Progress on the pragmatic front is even slower and harder than in semantics, mostly because our understanding of human pragmatic capacities is still in its cradle. When we consider the reproduction of pragmatic abilities on the computer we encounter the same problem described above with respect to semantics: it is not at all clear what our doing the various things that we do with language exactly amounts to, and therefore it is not clear when a computer would count as doing these things. This problem can be circumvented in the case of pragmatics as it is in semantics: we satisfy ourselves in trying to create on the computer syntactic processes that will have outputs that match those of our own cognitive processes. Here are two examples, two domains of AI research that consist in attempts to reproduce human pragmatic capacities on the computer. These examples can be described as respectively belonging to the two domains of pragmatics defined in 2.3 — Psychopragmatics and Sociopragmatics — and in both of them progress has all but bogged down so far, much before success could be announced.

The first domain is that of *expert systems*: computer programs that are supposed to reproduce human *reasoning* in solving problems from some restricted domain. (Reasoning includes deduction, i.e., logical inference, but goes much beyond it; therefore, our claim here that reasoning in language is part of pragmatics is consistent with the claim [made above] that inference is essentially a semantic capacity). A program of this kind has a knowledge base, in which a body of relevant data is stored; usually these data are represented in some formal language, i.e., in a way that approximates our own linguistic representation of the same data. Given a problem from the domain in question the program reasons (syntactically) on the basis of the data that it has, and comes up with an answer.

Expert systems had some initial successes in dealing with some very narrow domains; a famous example is that of MYCIN (Charniak and McDermott 1985: 465-468), an expert system that diagnoses quite well what type of blood infection a patient suffers from, on the basis of laboratory tests that the patient undergoes. (The knowledge base of the program consists of a list of rules employed by human experts.) However, attempts to tackle somewhat wider and more complex domains of human reasoning ended with failure: the programs could not approximate (and certainly not improve) normal human reasoning capacities. It is not agreed upon what the exact reasons of this failure are (and what the prospects are for overcoming them), but arguably some of these reasons have to do with our only taking the first steps in Psychopragmatics, i.e., with our lack of ability to articulate what human methods of using language in reasoning are.

The second domain of AI research we consider can be said to originate from A. Turing's so called imitation game, which became known as the Turing Test (Turing 1950). In this test a computer is supposed to simulate the human ability to make conversation, and this to the degree that an interlocutor whose aim is to find out whether he is communicating (in writing) with a computer or with a human will not be able to do so. (The computer will pass the test if often enough it will be mistaken by an interlocutor for a human being.) Turing thought that once a computer passed this test it would count as intelligent, but his interpretation of the test need not concern us here; what is important for us is that the test requires that the computer display pragmatic abilities: in order for the computer to pass the test it would have to be able, e.g., to make (and understand) implications, understatements, ironic gestures, threats, promises and so on.

Turing was sure that by now (i.e., by 1999) computers would pass his test. This is not because he was optimistic about our ability to analyze human pragmatic abilities and reproduce them computationally; rather, to a large degree Turing was unaware of these abilities' existence. In fact, we now know that we are very far from being able to construct a machine (i.e., to write a program) that passes the Turing Test; what may seem substantial steps towards reaching this goal, e.g., the well known program ELIZA, which simulates a psychotherapeutic interview, are in fact nothing of the kind. (Those who mistake ELIZA for a human psychotherapist do *not* aim during the interaction with the program to find out whether it is a program or a human being. Therefore the program cannot be said to have passed the Turing Test, even if some people interact with it without being aware that it is a computer program and not a human.) As in the previous example, the reason for this shortcoming is to a large degree the great need that we have to make progress in pragmatic research (this time in Sociopragmatics): until we understand better the working rules according to which we use language in communications the chances of making the computer mimic us are slim.

d. The situation in phonology is somewhat similar to the above described state of affairs in semantcs and pragmatics (see Dreyfus 1992). That is, there had been several initial successes in the domain of artificial voice recognition, and slow progress has been made ever since. However, the optimistic predictions made a few decades ago, to the effect that we would have by now computerized agents that can match human phonological capacities, have failed. Humans can recognize with practically no errors expressions uttered by different speakers in widely different contexts, while computer programs are so far much more limited: they are often capable to recognize only the voice of a specific speaker in favorable circumstances. Part of the reason for this gap between man and machine in what could be naively thought of as a straightforward task may be this: human phonological performance relies substantially on semantic and pragmatic co-textual and contextual considerations, and hence the above described limitations on the computational simulation of such considerations hinders the computer also in what one might thing to be the more menial phonogogical tasks.

In an attempt to overcome this and other obstacles to human-computer communication, some researchers have been developing computerized agents that do not rely only on language (written or oral) but also on the production and perception of visual and other bodily signals (cf. Cassell et al., eds., 2000). Although enthusiasm for the first achievements of these «Embodied Communicative Agents» is contagious, it remains to be seen whether the story of initial successes followed by very slow progress that we have observed in syntax, semantics, pragmatics, and voice recognition, will not repeat itself in this new front too.

2. The Digital Information Age and Natural Language Processing

2.1 Computers, Communications, and the Information Age

Since the mid-eighties the uses that computers are put to and our expectations from them have undergone a significant shift. Admittedly, computers are still used for «number crunching» in physics laboratories and in banks, and so called classic AI research still goes on, albeit more realistically. However, a host of new uses of the computer are now given a leading role: computers everywhere on the globe are used for the storage, presentation, transformation and transmission of *information*. All major types of information — text, picture and sound — have proved to be amenable to digitization, and thus open to computational processing. This fact has important ramifications for the use of computers in the handling of information and its communication.

First, there is the simple issue of storage capacity. Ever bigger quantities of digital information can be stored on increasingly smaller devices, and thus people can have in their laptops complete encyclopedias, or complete galleries of pictures.

Second, the interface between humans and the computer, once of minor interest, is becoming a point of central concern. Information is presented now by the computer graphically, acoustically, and through Virtual Reality technology (the recreation of three dimensional reality on the computer screen); this is made possible by the growth in the storage capacities and computation speed of computers. Similarly, the cues that the computer can take from its human users are getting more and more diverse.

Third, as data are represented by the computer digitally they can undergo all kinds of manipulations and transformations: they are malleable as never before. One major consequence of this new malleability of information is the introduction of *multimedia*, whose underlying idea is this: if textual, visual and auditory kinds of information are stored and processed by the computer in similar ways (i.e., digitally), then the traditional barriers among these media fall down, and one can mesh them together. Multimedia documents are bodies of information in which such meshing together is carried out (in ways that are richer and more complex than what is done in illustrated books or subtitled films, which are primitive multimedia documents).

Finally (and probably most importantly), the digital clothing of information opens up new ways for its communication, the foremost of which is the Internet. The Internet includes the World Wide Web, a network consisting of a huge number of information sources (servers) which anyone can log in into and browse through on-line; also, the Internet enables direct contact among any number of people, e.g., according to their common interests. The existence of a network of such unprecedented complexity is made possible through the digital tagging of every node of the network (be it an information source or a browser), as well as every piece of information moving through it, and by tracking and handling all this symbolic transport (on the basis of its tags) through the use of computational resources that are distributed over the network.

The Internet incorporates within it the mass media, the telephone and the marketplace: anyone who uses it can address all kinds of groups of people, from a single person anywhere on the globe to the whole network. Due to this power the Internet has important ramifications for all domains of human life, such as the economy, education, and politics. Indeed, because so much interpersonal interaction can take place on it, the Internet is commonly viewed as creating a new space — Cyberspace — in which a great part of human life can go on.

2.2. Natural Language and the Computer as an Agent, a Tool, and an Environment

What are the implications of these developments in computer technology for the attempts to reproduce natural language processing on the computer? Are these attempts directly relevant to the role of the computer in the information age, or is computerized natural language processing expected to be of declining interest in this age? At first blush it would seem as if the latter of these two options — that of declining interest in computerized linguistic processing — must hold. In order to see why, let us consider the developments described in the previous subsection in a slightly different light, using some old fashioned, non-technical terminology.

We humans view ourselves as *agents*. That is, we are creatures that have a will, and who act upon their desires; we are active. Our actions take place in a passive *environment*, to which we do not ascribe agency: we do not think of the inanimate objects in the world around us as having a will, and as having desires that are acted upon; rather, these objects are viewed by us as passive entities, behaving according to pre-established regularities. In between our active selves and the passive environment are located our *tools* — parts of the environment that we use in order to advance our goals and to carry out our actions. Tools themselves are passive, but their being used by active agents for some purpose is constitutive of their being tools.

These three terms — agent, environment, and tool — can be now used to mark distinct stages in the evolution of our view of the computer, and (consequently) of our using it. Roughly speaking, we (humans) started our acquaintance with the computer by thinking of it as an *agent*, then moved to thinking of it as a *tool*, and are now on the verge of viewing it as underlying a new *environment*. Briefly, here is why.

The first part of the claim is supported and substantiated by the content of the first section of this paper. The computer was described there as a machine that carries out computations — something that we, as active agents, do. The AI attempts to reproduce various of our cognitive abilities on the computer are in the same vein; underlying them is the view of the computer as a mechanized agent. This agent may be programmed to help us with some things that we wish to accomplish, but it (the computer) would help us like an agent — a friend, maybe, or a slave: it is given a task, and it is supposed to return with the result. This is how batch programming (as opposed to interactive programming) works: the computer is given some input, and is then left to its own devices, so to speak, in coming up with the right output. Also, the work on robots, which goes hand in hand with AI research, embodies (literally) this approach to computers.

The second conception is that of the computer as a tool. This conception evolved with the development of graphic interactive programming — computer programs that are constantly used during a certain process, and that therefore do not have a clearcut beginning and end to their execution. Word processors are typical examples of such programs, and so are CAD/CAM programs — programs for computer aided design and manufacturing. These programs are indeed used like tools, and the (heavy) computations they perform have little or nothing to do with the simulation of cognition. For example, consider the difference between the use of an AI program in the design of some building or machine, and the use of a tool-like program in the performance of such a task: an AI program would simulate some of the considerations of the designer, while a CAD (tool-like) program would simply help the designer make these considerations on his own, e.g., by presenting to him visually the various options that he must choose from at each stage of the designing process.

Finally, the evolving uses of the computer in communications — especially the Internet — lead us toward viewing the computer not as an agent or a tool, but rather as underlying an environment in which human interaction takes place. Admittedly, we use the computer in communications as a *tool*, like the pen or the telephone, but as this tool develops and encompasses an ever growing part of our activities, there occurs a change of perspective: we turn to view ourselves as communicating with each other *through* a medium that is put in place by computing processes, as we communicate through ordinary three-dimensional space. This change of perspective is most evident in the term 'Cyberspace', but it is not merely a matter of terminology. Rather, the evolution of the computer from a tool into an environment witnesses the sharp increase in its importance to our lives; this was the case for the book, the building, and the road, which are viewed by us as parts of our environment, and it is also the case for the computer network. (This view of the computer can be said to have been anticipated by Heidegger, and to some extent by Wittgenstein, who thought of *language* as an environment: the former in claiming that language is 'the house of being', and the latter in establishing a close connection between 'language games' and 'forms of life'.)

The way all this is relevant to the question we started with should be clear. The transformation just described in the way we view the computer should *diminish* our interest in computerized natural language processing, because linguistic capabilities are something that is required from an agent, not from a tool, and certainly not from an environment. If we talk with each other *through* the computer, so to speak, and not *to* it, then the computer need not have any speech processing abilities.

A look at the early literature concerned with the Internet and other information technologies reveals that indeed there is surprisingly little concern in it with natural language processing issues in particular, and with AI in general. The new cycle of computer technology is seldom viewed by its leaders and prophets as a continuation of the previous, computer-as-agent cycle; rather, it is quite often presented by these leaders and prophets as a step in a new direction.

Is there indeed such lack of dependence between the different cycles of computer technology? Is the information age in no need of research into human natural language processing, and does it not depend in any way on the implementation of the fruit of such research computationally? In the next subsection we see that this is not the case.

2.3. Natural Language Processing in the Information Age

We turn now to three examples of the way in which information technology does, in fact, depend on natural language processing. These examples prove the assessment presented in the previous subsection to be misguided, and after going through the three of them we conclude by stating why this assessment is indeed wrong.

a. *Semantic Search.* The Internet makes available to us huge quantities of information, stores of data much bigger than anything we have ever seen before. Now these seas of data can be of use to us only if we can fish from them information that is relevant to our needs and interests; having useful information at the tips of our fingers is of no great significance if this information is buried under an avalanche of useless data.

But can we easily find our way in Cyberspace and fish for relevant information? Well, we humans cannot. It is beyond human powers to go through millions of Web sites, for example, and see if their content bears upon a certain subject matter. (Note that this is as opposed to our ability to take in through our senses huge quantities of data about the world around us, and then pick out at any given moment what is important for us.) So does the Internet consist in a flood of useless information? No, it does not. As anyone who has used the Internet knows, there are computer programs, called Search Engines, that go through Cyberspace for us, and return to our screens with information on the topics we want to look into. Give a search engine such as Altavista or Yahoo the input string 'San Francisco,' for example, and these programs will find for you (i.e., give you the addresses of) files of information on that beautiful city.

Note now the following important point: almost all existing search programs are essentially *syntactic*. That is, when a search engine is given some string of symbols as input it simply looks for that string (inside some Web sites, or in Web sites' names, or in some previously compiled directory); the engine does not take into account any semantic aspects of the string in question. We humans, on the other hand, definitely do take into account meaning when we look for information that is relevant to a certain topic. *Relevance is a semantic notion*, and therefore relevance connections among linguistic expressions clearly depend on what these expressions mean. Therefore what search engines do is essentially different from what we humans do when we search for information, e.g., when we go to the library, ask our friends, or 'go through' our own memory: we appeal to semantics while current search engines do not. (Note that we too definitely use syntax, e.g., when we look for a word in the dictionary; however, we are not limited to the level of syntax.) This difference between the two modes of search gives ample reason to think that computerized syntactic searching will not be able to match its semantic human counterpart.

As a concrete example, consider a hypothetical search for information about San Francisco, one that takes into account semantics. For a start, note that 'San Francisco' is a name of a place, as opposed to 'Socrates' or 'Communism'; therefore, looking for San Francisco's location in an *atlas* or a *map* makes sense (as opposed to looking for Socrates or Communism in these information sources). As a place San Francisco has its weather, which could be looked up in *weather* information; Socrates and Communism, on the other hand, do not have weather. San Francisco is a place of a specific kind — it is a city; therefore it has streets, which could be looked up in a *street map*, and buildings, which could be the subject matter, e.g., of information sources on *architecture*. Furthermore, as an American city San Francisco's history is part of *American history*; inside the city's history, for example, there could be located the *gold rush* era and the erection of the *Golden Gate Bridge*. These semantic connections can be tracked further and further, but the point should be clear: the links among the italicized strings in this paragraph ('San Francisco', 'atlas', 'map', 'weather', 'street', 'architecture', 'American history', 'gold rush' and 'Golden Gate Bridge') are links of meaning, and therefore only a search process that takes meaning into account can trace them and follow them.

The upshot of the previous two paragraphs can be summarized. Naive, syntactic search for information (in the Internet, as well as other media) cannot reproduce the results of the semantic procedure that we humans perform. Therefore if we stick with syntactic search methods we shall most probably fall short of fully tapping the resources that information technology makes available to us. If, on the other hand, we make progress in understanding human semantic processing and in reproducing it on the computer, then we stand a better chance to fish successfully in the newly established seas of digital information. (See also Cheng and Wilensky 1998.)

Let us quickly address two objections to this conclusion. One is that the example given above (concerning San Francisco) does not show in a convincing way that semantic considerations are indeed required in information search. It could be argued that a mere syntactic search for the string 'San Francisco' would surely bring forth an information file on the city, such as an encyclopedia entry, in which all the above mentioned semantic connections have been made (connections concerning, e.g., location, weather, and history); therefore the search program itself need not reproduce these connections.

The answer to this objection is twofold. First, in the case of the city of San Francisco the computerized search may indeed piggyback on previously done human work (in which semantics was obviously taken into account). However, this is not the case as regards places that are not tourist destinations, and even less so as regards other things and concepts.

In the general case no ongoing human information accumulation is performed, and therefore there is no replacement for the computer's being able to make such semantic connections as that between the notion of a place and maps or atlases (as information sources).

Second, and more interestingly, the kinds of information search that will be called for in the information age are extremely complex, user specific, and transitory in character, and for such kinds of search the 'encyclopedia entry' option envisioned by the objection simply does not exist. For example, consider a personalized program that is supposed to survey for me various television channels and information sources on the Internet and put together a daily bulletin of news that matches my own idiosyncratic needs and interests at any given time. Clearly such a program (or an *agent*, as these programs are often called) will not be able to rely on previously compiled bodies of data; it will have to make semantic relevance considerations on its own.

A second objection to our conclusion is this. In order for the computer to make semantic considerations it has to go beyond syntax, but it cannot; all it is capable of is syntactic manipulation. (J. Searle has forcefully argued to this effect, on the basis of his Chinese Room thought experiment.) Semantic search is beyond the scope of the computer's ability, and therefore the suggested computerized semantic search is a contradiction in terms.

The answer to this objection has already been given in the first section of this paper. We can put aside the deep philosophical and scientific issues that semantics brings forth, and aim at simulating computationally (i.e., syntactically) processes that are inherently semantic for us humans. Thus it is sufficient for us that the computer would be able to fish for us syntactical objects (i.e., digital information files) that *we* would judge as relevant to our interests; we need not bother ourselves with the question whether the computer does or does not understand what it has done and what it came up with. (See 1.4 clause [b] above for further elaboration of this answer.)

b. *Data Mining.* Having dealt with an example from semantics at some length, let us merely point out two further examples, from the domain of pragmatics.

First, an example from Psychopragmatics — the use of language in thought. Notice that in the previous clause we have considered merely the *search* for information that is relevant to some topic; we have not said anything regarding what is supposed to be done with the information that has been found. Now information is usually supposed to help us reach conclusions, in the form, e.g., of practical decisions and scientific hypotheses. Therefore the following question can now be asked: Can we humans make use of the information that will be available to us in the digital age, even if the information that is relevant to our needs is brought before us? That is, can we reach reasonable conclusions on the basis of this information?

As in the previous discussion, the answer to these question seems to be negative, at least in part. The rapid growth of accessible information leads to a similar growth in the number of cases in which information analysis is required, and also to a growth in the magnitude of the body of information that has to be dealt with in each such case. Human capacities are simply not sufficient for such data processing.

As in the previous discussion, the desirable solution is to reproduce our conclusion-making abilities computationally. The area of computer research in which such reproduction is attempted is called Data Mining — the computerized processing of raw data and the extraction of knowledge from them, in the form, e.g., of hypotheses or decisions. Much of the data that Data Mining is concerned with are formulated linguistically, and therefore progress in this area depends to a large degree on our having better understanding of how we humans use language in thinking — i.e., on progress in Psychopragmatics. (There is growing awareness of the dependence of computerized thought simulation on linguistic processing, even outside the academic ivory tower; for example, note the following title of a recent advertisement by Mercedes Benz in *Time Magazine:* 'Language is the picture and counterpart of thought'.)

c. *Human-Computer Interface.* Finally, we get to Sociopragmatics — the use of language in communication. The main way such language use arises in the information age is, in a nutshell, this: the more elaborate and flexible the uses we want to put the computer to are, the greater our need to communicate with it in our own natural means of communication, i.e., in natural language. Here is why.

We have seen that in the information age there will be a growing use of the computer as a medium of communication with other people, and as a flexible means of access to various kinds of information. For example, we are soon expected to use the computer to contact our friends, our bank, and the supermarket, and to get through it political news on the issues that matter to us, economic news that concern our interests, and maybe gossip having to do with our favorite celebrities. In order to get all this from the computer we must constantly program it — we must tell it what to do. Using anything but our own natural language (or some close approximation of it) for this purpose will be neither practical nor wise, for it would involve translating our natural language utterances, which are often syntactically deviant and incomplete, into precise and univocal formulae, designed with a limited number of purposes and domains in mind. Thanks to the flexibility permitted by sociopragmatic principles of communication, we are able to convey our intentions and understand those of our interlocutors without undertaking such a painstaking translation task, and without submitting to the artificial constraints of pre-formalization. Except for a few (albeit important) domains, formal languages have been of remarkably little use in communication: hundreds of artificial languages have been devised for this purpose, and none has managed to establish itself in any significant way (Eco 1995).

We need our computers to be competent natural-language speakers, then, and hence the study of Sociopragmatics and the computational simulation of human pragmatic abilities turn out also to be of major importance for the development of information technology.

* * *

It is natural to ask now the following question: how can the last two subsections be reconciled? According to the analysis presented in subsection 2.2 the role of computerized natural language processing is expected to decline in the information age, and yet in 2.3 there has been presented ample proof that this is not the case. So where, exactly, do the observations in 2.2 go wrong? We conclude by answering this question, using one of the major notions introduced in subsection 2.2, i.e., that of agency.

As said in 2.2, the computer evolved during the past few decades from a would-be agent into a machine that underlies a new environment. Now it turns out that this new environment is different in various ways from the one we are used to, and that therefore it is not as easy as expected for us to accomplish in it what we want to — e.g., to gather and use information. It is for this reason that computerized agency reappears. In the information age we need computerized agents, but not in order to act on our behalf in the real world, for which purpose we are well adapted; rather, we need them for the purpose of representing us in Cyberspace, which in some ways is alien to us, as we need the Mars Explorer to represent us on Mars.

Together with the resurrection of computerized agency we find again need for computerized linguistic processing: Our emissaries in Cyberspace must perform (simulated) reasoning and they must be communicated with, and for these two reasons they must be endowed with linguistic abilities.

References

AUSTIN, J. (1962). How to Do Things with Words. Oxford: Clarendon Press.

BARWISE, J. & COOPER, R. (1981). Generalized Quantifiers and Natural Language. *Linguistics and Philosophy* 4: 159-219.

BOOLOS, G. & JEFFREY, R. (1974). Computability and Logic. London: Cambridge University Press.

CASSELL, J.; SULLIVAN, J.; PREVOST, S. & CHURCHILL, E. (eds.) (2000). Embodied Conversational Agents. Cambridge, Mass.: MIT Press.

CHARNIAK E. & MCDERMOTT, D. (1985). Introduction to Artificial Intelligence. Reading, Mass.: Addison Wesley.

CHENG, I. & WILENSKY, R. (1997). An Experiment in Enhancing Information Access by Natural Language Processing. Computer Science Department Report CSD 97-963, University of California at Berkeley.

CHOMSKY, N. (1965). Aspects of the Theory of Syntax. Cambridge, Mass.: MIT Press.

CHOMSKY, N. (1980). Rules and Representations. New York: Columbia University Press.

CHOMSKY, N. (1988). Language and Problems of Knowledge: The Managua Lectures. Cambridge, Mass.: MIT Press.

DASCAL, M. (1977). Conversational Relevance. *Journal of Pragmatics* 1: 309-327.

DASCAL, M. (1992). Why Does Language Matter to Artificial Intelligence? *Mind and Machines* 2: 145-174.

DASCAL, M. Forthcoming. Hobbes' Challenge. *Minds and Machines*.

DEEMTER, K. van & PETERS (eds.) (1996). Semantic Ambiguity and Underspecification. Stanford: CSLI.

DREYFUS, H. (1992). What Computers Still Can't Do. Cambridge, Mass.: MIT Press.

ECO, U. (1995). The Search for the Perfect Language. Oxford: Blackwell.

FODOR, J. (1994). The Elm and the Expert. Cambridge, Mass.: MIT Press.

GULLVÅG, I. & NAESS, A. (1995). Vagueness and Ambiguity. In: DASCAL, M. et al. (eds.), *Philosophy of Language: An International Handbook of Contemporary Research, vol. 2*. Berlin: De Gruyter.

GRICE, P. (1975). Logic and Conversation. In: P. COLE & J. MORGAN (eds.), *Syntax and Semantics*. New York: Academic Press.

JACKENDOFF, R. (1983). Semantics and Cognition. Cambridge, Mass.: MIT Press.

JACKENDOFF, R. (1987). Consciousness and The Computational Mind. Cambridge, Mass.: MIT Press.

KAMP, H. (1981). The Paradox of the Heap. In: MÖNNICH, U. (ed.), *Aspects of Philosophical Logic*. Dordrecht: Reidel.

LEWIS, H. & PAPADIMITRIOU, C. (1998). Elements of the Theory of Computation. Upper Saddle River, New Jersey: Prentice Hall.

MORRIS, C. (1938). Foundations of the Theory of Signs. Chicago: The University of Chicago Press.

PINKER, S. (1995). The Language Instinct. New York: Harper.

PUSTEJOVSKY, J. (1993). Semantics and The Lexicon. Dordrecht: Kluwer.

REICHMAN, R. (1986). Getting Computers to Talk Like You and Me. Cambridge, Mass.: MIT Press.

SEARLE, J. (1969). Speech Acts. Cambridge: Cambridge University Press.

SEARLE, J. (1984). Minds, Brains and Science. Cambridge, Mass.: Harvard University Press.

SHOENFIELD, J. (1967). Mathematical Logic. Reading, Mass.: Addison-Wesley.

SPERBER, D. & WILSON, D. (1986). Relevance: Communication and Cognition. Cambridge, Mass.: MIT Press.

TURING, A. (1950). Computing Machinery and Intelligence. *Mind* 59: 433-460.