MAC
CONVER

MACHINE CONVERSATIONS

edited by

Yorick Wilks
University of Sheffield
United Kingdom

SPRINGER SCIENCE+BUSINESS MEDIA, LLC

Library of Congress Cataloging-in-Publication Data

Machine conversations / edited by Yorick Wilks.
p. cm. -- (Kluwer international series in engineering and computer science ; SECS 511)
Includes bibliographical references and index.
ISBN 978-1-4419-5092-5 ISBN 978-1-4757-5687-6 (eBook)
DOI 10.1007/978-1-4757-5687-6
1. Human-computer interaction. 2. Natural language processing (Computer science) 3. Artificial intelligence. I. Wilks, Yorick, 1939- . II. Series.
QA76.9.H85M33 1999
004'.01'9--dc21 99-30264
CIP

Printed on acid-free paper.

Contents

Preface

Human-computer conversation is a technology that has reached the same stage of development as some better-known areas of language processing by computer, like machine translation (MT) and information extraction from texts (IE): it is an area of natural language processing (NLP) technology where new and striking developments come from effectively ignoring received theories and wisdoms. It is a truism in science as a whole that this can occur, but in artificial intelligence (AI), of which NLP is part, this often takes the form of striking out towards a practical implementation of a difficult task, with all the risks that such course of action entails. Recent examples would be the brief flowering of statistically based MT at the end of the 1980's, and the rapid growth of the IE movement at about the same time, a growth that continues.

Both those movements effectively ignored the current state of theory and any search for the foundations of the subject, in favour of what one could call having-a-go. This is an old AI tradition, sometimes called throwaway-AI, or more charitably rapid prototyping. It is widely condemned by theorists of all types, who point out yet again that climbing a tree cannot get one to the moon, even if it feels like a step in the right direction. Interestingly, too, both those movements were associated with the development of strong evaluation regimes, imposed by the sponsors of much of the work. Rapid improvements over time were seen, but then they began to fall off and it was again argued how much constant evaluation can stifle theoretical creativity and novelty, all of which is certainly true, though the criticism fails to capture how little such theorists had been interested in systems that were evaluable at all until this new pressure appeared.

Machine conversations with humans, as an area of research and development, shares another striking property with machine translation: both made rapid advances about twenty-five years ago and some doubt that the amount of theoretical and practical effort expended since have changed the situation very much. In MT it is clear that SYSTRAN, now well into its fourth decade, is still the most used system for large scale tasks and the one that still wins competitions against those embodying newer, more perspicuous, approaches.

In machine conversation, Colby's PARRY, devised at Stanford in the early Seventies, set a standard that has not been reached since, at least until recently. PARRY's position was, for most observers, hidden behind the greater publicity achieved by Weizenbaum's ELIZA, a much inferior conversationalist. Also, the fact that PARRY was designed to model a mental disorder (PARanoia) made it easier both to excuse its lapses and to dismiss its general significance for the area.

Four features of the machine conversation problem have brought us to the situation that motivated the essays in this book. First, dialogue analysis has now gathered a great deal of research attention (within linguistics proper and within AI-related work) under such titles as conversational analysis, dialogue acts, speech acts, conversation acts and so on, but virtually none of such work has contributed to any tradition of building real performing and robust conversational partners. That is a remarkable fact, though one not unknown before in the history of language processing, as I noted earlier. It is certainly a situation to make one sit back and ask calmly what has been the point of all that work if it has so few evaluable outcomes over twenty five years.

Secondly, one notes the resurgence within computational linguistics of empirical methods, methods driven by linguistic data (e.g. Charniak, 1993; and Wilks, 1996). This movement is now spreading to complex areas like language pragmatics and human conversation, partly (within the English language at least) because of the availability of the dialogue part of the British National Corpus, amounting to some two million words. One of the systems described here has built structures in part from analysis of those interchanges, and indeed from the corpus comprised of all the past Loebner competition dialogues (see below). Well-motivated empiricism, in addition to stamina and application, now seem to offer possible ways forward in this area and that is reflected in some of the papers here.

Thirdly, the need for such conversational agents has become acute with the widespread use of personal machines with which to communicate and the desire of their makers to provide natural language interfaces. There is good reason to believe that most, if not all, major hardware and software houses are working on some form of this problem, but the literature in the area shows little of proven practical help, only a welter of claims. Indeed, below the academic surface, work has continued in many places on such agents, developing a sort of craft-skill in how to program them, one that owes a great deal to AI conventions and to the approach embodied in Colby's PARRY. (Parkison et al., 1977) This is a theme developed in the book, but it can be summarized as: machine conversationalists have to have something to say, have to simulate having a story to tell, rather than just reacting to, disambiguating or otherwise processing what they receive.

For many this is a triviality, and the hard part is going from that to a craft skill that can express it in a program. I write "craft skill" in opposition to forms of theory that constantly reappear as new claims but with no method at all of tying the observation to any performance. Most recently, Poesio and Traum (1997) have developed a new fusion of Rochester speech-act techniques with an Edinburgh style of discourse-representation-theory, leading to a complex analysis of an interchange like:

A: There is an engine at Avon.

B: It is hooked to a boxcar.

which provides, over many pages, an analysis and representation capable of referring the "it" to the engine, as if there were serious alternatives, and then going on to talk of linking all this to a theory of what they call "conversational threads", which they concede was present many years ago in Schank's notion of script (Schank and Abelson, 1977). To anyone actively interested in machine conversation, all this is remarkable: the example dialogue is both artificial and unmotivated—why does B. say what he does?—and the linguistic pronoun issue has been a non-problem for a quarter century. The real problems, like conversational threads, however titled, are left always to some future time. Meanwhile, others, some of them represented in this book, are actually engaged with the construction of such conversation programs and the skills to develop them in a general and robust way. By way of historical analogy, some may recall that, only ten years ago, papers on computational syntax would spend pages discussing the parsing of a sentence like "Uthor sleeps", which could pose no conceivable problem; all such discussion has been swept into oblivion by the establishment of realistic parsing of real sentences evaluated in competitions like PARSEVAL.

The history of AI has shown the value and importance of such craft skills in all its areas from games to vision to language: the discovery of the underlying structures and principles to make something work, by experience. All formalisms and searches for justifiable *Anfangspuenkte* never replace this phase, and rarely add much later when tidying and refining is required. What they do is continue to is endlessly discuss the same examples in different forms as if the problems they present had not been cleared up decades before.

Fourthly and lastly, one must mention the role here of the Loebner competition, which has played the role of an evaluation process in machine conversations that DARPA has played in MT and IE. That an amateur with flair, with a notoriously eccentric personal web page, started this competition, rather than a staid government agency, has discredited it for some. Nonetheless, the Loebner competitions on human-machine conversation have been well conducted by the ACM and panels of experts and have certainly increased attention to, and activity in, the craft skills, and the 1997 Loebner winner is represented in this volume.

The drawback of the competition (and here it differs from MT and IE DARPA competitions) is that entrants bias their systems towards pretending to be people (it being the job of the competition judges to sort the people from the programs) and that is not necessarily a feature one will want in the future in useful machine conversationalists that are not pretending to be other than they are!

The papers in this volume came from an attempt to gather a collection of the best work available in the practical arts of machine conversation: not with the desire to exclude any theoretical claims but to keep the focus on the task and on performance. This had the effect of showing some of the first rate work going on in industry, quite apart from the academic tradition; it also brought out striking and relevant facts about the tone of machine conversations and what users want, and focused on non-linguistic aspects of these conversations that we, in the typing business, tend to ignore. Above all, it showed again the pioneering role of Ken Colby at Stanford in the 1970s, and this book is dedicated to him.

References

Charniak, E. 1993. Statistical Language Learning. MIT Press, Cambridge, MA.

Parkison, R.C., Colby, K.M. and Faught, W. 1977. Conversational language comprehension using integrated pattern-matching and parsing. Artificial Intelligence, 9: 111-134. Reprinted in Readings in Natural Language Processing, Grosz, B., Jones, K.S., Webber, B.L. (Eds.) (1986), Morgan Kaufman Publishers, Inc., Los Altos, CA.

Poesio, M. and Traum, D. 1997. Representing Conversation Acts in a Unified Semantic/Pragmatic Framework, In preprints of the AAAI Fall Symposia, Stanford, CA.

Schank, R. and Abelson, R. 1977. Scripts, Plans and Goals. Erlbaum, Hillsdale, NJ.

Wilks, Y. 1996. Special Issue of the Communications of the ACM on Natural Language Processing. January 1996.

Acknowledgements

Considerable thanks are due to David Levy, who thought of staging the original event that gave rise to this book at the Grand Hotel Villa Serbelloni at Bellagio, Italy, one of the world's finest spots, as Pliny knew long before us when he built his villa there. Also for substantial help and advice on the programme etc. from Louise Guthrie, and for great help with organizing from Gill Wells. An enormous debt of gratitude is due to Gillian Callaghan for much painstaking work on this volume. And to Roberta as always for everything.

Yorick Wilks
Sheffield

1 Dialogue Programs I have Known and Loved Over 33 Years

K.M. Colby

In the mid-1950s, when I first read about computers, it occurred to me that they might be used to provide some form of talk-therapy for the thousands of mentally ill warehoused in large mental hospitals receiving only custodial care. I envisioned some sort of large time-shared system running programs that could be communicated with in everyday natural language. But as we all have learned, it is a long way from an idea to its implementation.

In the late 1950s, Allen Newell paid me a visit. At the time I was practicing psychiatry in Northern California and trying to hand simulate a neurotic process with boxes of cards. Allen encouraged me to learn a programming language (IPL-V) and to get more involved in AI research. I decided to get into full-time research which meant academia. So after a year at the Center for Advanced Study in the Behavioral Sciences in Palo Alto, I joined the psychology department at Stanford and soon the Computer Science Department there.

And then I met Joe Weizenbaum. He was working for General Electric having contributed to the IRMA program which handled checking accounts for the Bank of America. We met through Ed Feigenbaum and even considered forming a company called Heuristics Inc. along with Ed and John Gilbert, the statistician at the Center for Advanced Study. Joe and I spent many hours together discussing the problems of talk-therapy programs in natural language. To change its mind, I was trying to communicate with my neurosis program using a form of Basic English, but it was hopelessly cumbersome.

Joe moved to MIT where, with the aid of the time-shared system MAC, he wrote the first up-and-running dialogue program ELIZA with me contributing many of the admittedly limited input responses characteristic of talk-therapy conversation. The first description of ELIZA appeared in the Harvard Review and was titled “Conversations With a Mechanical Psychiatrist”. We soon had an ELIZA-like program running at Stanford called the “MAD DOCTOR”—not because it exhibited anger but in the British sense of the term “mad” meaning mentally ill. I saw the program as having a potential for psychotherapy.

But Joe did not. He objected strongly to using computers for therapy. He even said he was “shocked” by the publication of our paper (with James Watt and John

Gilbert) on the subject. In 1965, before publication, we sent Joe a copy of this paper and he requested that he write the section on ELIZA which we granted. It is hard to imagine how he could be shocked at a paper part of which was written, word for word, by himself. But Joe was not attacking just us. In a 1976 book, he objected to much of AI work as "immoral", "obscene", etc. My own interpretation is that in part he was striking back for being rejected by the AI group at MIT.

As an amusing aside, when Joe visited the Stanford area in the 1960s, he would often stay at our home - this of course before his anti-AI blasts. Our son Peter would give up his room to accommodate Joe. When Peter was a student at UC Santa Cruz, Joe gave a talk there. Afterwards Peter introduced himself to Joe and reminded him of how he had to give up his room on Joe's visits. In concluding their chat, Joe said to Peter "and give my regards to your mother". (!)

At the same time we were working on the talk-therapy program, I was interested in using programs to stimulate speech in non-speaking autistic children. In collaboration with Horace Enea and David Smith, we designed and ran many language-oriented programs which could hardly be called "dialogues" since they were quite rudimentary at the level of single words or short phrases. Some of the children began talking in response and some did not. These efforts have been taken up by Jill Lehman at Carnegie Mellon using the many advances in technology now available - animation, speech recognition, etc.

The next dialogue program was PARRY, a simulation of paranoid thought processes. Instead of having a program talk like a therapist, the goal was to have one talk like a paranoid patient. What we had learned from ELIZA and the MAD DOCTOR was that it might be possible to conduct a conversation using only word-meaning and that complex ontological designs (tree-structures categorizing world-knowledge with THING at the top and DOG at the bottom) were not necessary. Before he received his doctorate in linguistics, I had hired Roger Schank to work on the natural language problem. Roger worked with both Larry Tesler and David Smith but couldn't get along with either. He and I didn't see eye to eye about the usefulness of grammars in conversational language, so I let him go his own way in developing a conceptual dependency grammar. My main helper on PARRY was Sylvia Weber Russell who wrote the algorithm in LISP while I constructed the semantic data-base.

PARRY was designed to converse in the highly constrained situation of a first diagnostic psychiatric interview. As do human paranoids, he had a story to tell about how he was being persecuted, especially by the Mafia. If any of his "flare concepts", for example, were activated, he would reveal something about the Mafia. Hence any reference to gambling (or even Italy) would evoke part of his story. The importance of this point for machine conversation is that the participating program should have some sort of constraining anchor in the world so that its virtual person, or persons, can give direction to the flow of talk exchanges.

At Stanford, I had a Career Research Grant to do whatever interested me, but when President Nixon decided to abolish this luxury in the early 1970s, I realized I had to get a regular job and so accepted a professorship in psychiatry at UCLA. I brought William Faught and Roger Parkison with me and they both received their Stanford doctorates based on PARRY.

At Stanford, we had conducted a number of indistinguishability tests on the model with the help of Frank Hilf, a psychiatrist, and Hans Moravec, an AI graduate student. At UCLA we continued these tests and collected linguistic data from over 100,000 PARRY interviews on the ARPANET. Just when we were ready to try to "treat" PARRY by cognitive therapy, the project was defunded. (There is still no treatment for the paranoid mode). NIMH decided not to put any more money into AI. Such are the vagaries of grant funding.

In the mid 1970s an economist and population expert, Julian Simon, sent me a manuscript he was trying to get published describing his own depression and how he overcame it. He wondered if I could model depression as was done for paranoia. Still pursuing talk-therapy, I responded I would be more interested in developing a cognitive therapy program for depression. Roger Parkison and I had written a primitive Joe-the-Bartender program that could converse superficially about the weather, sports, etc. as bartenders do. While waiting for Simon's book to get published, son Peter and I developed the program 'Overcoming Depression' with a text mode based on Simon's work and a dialogue mode based on our own heuristic methods for making sense out of the tangles of conversational language in the interpersonal domain. Simon's book was finally published as Good Mood in 1993 and since then we have given away free over 1300 copies of the program to buyers of the book. Sad to say, Simon died recently. We will always be indebted to him for his emotional and financial support in our effort to make an end run around the social stigma associated with depression.

Like most prophecies, my initial vision of computer-based therapy was far off (even John McCarthy predicted we would be out of gasoline by the year 2000) because I did not anticipate the invention of personal computers in which a sufferer can treat himself at home with the help of a talk-therapy program. Because of innovation, it is difficult to predict - especially the future.

Risking prophecy again, I do not think computers as conversational agents have much of a future in psychotherapy. But they may have a great future commercially because people enjoy conversing with computers that tend to say odd but engaging and pertinent things. I view our dialogue mode as somewhat analogous to the Wright brothers' airplane. Once we get it off the ground (it now makes sense about 90% of the time), we can then worry about serving 150 people the shrimp cocktails at 600 miles per hour 35,000 feet above the earth.

Hence we continue to work on it every day.

2 Comments on Human-Computer Conversation

K.M.Colby

It was heartening to see so many men and women at the Villa Serbelloni eager to work on this seemingly insuperable and hence challenging problem of human-computer conversation. From the variety of issues, arguments, opinions, jokes, etc., I have selected a few that stand out in my memory. The order presented here does not reflect the order in which they occurred.

The term "theory" often appeared. There seem to be all sorts of theories. In physics, a "scientific" causal-explanatory theory consists of lawfully-related coherent abstract entities, with relevant variables specified and postulated about a system to account for the law-like generalization of its evidential phenomena. It states what a system is and what it must do. But can there be such a theory of conversational language which is convention-governed, providing room for choice, rather than physical-law governed? Since words and word-meanings are governed by conventions of linguistic communities, it is doubtful that a sound analogy with physical theories can be made. OED has recently added 3,000 new words to its vocabulary, which is not like discovering 3,000 new elementary particles. Conventions, as institutional facts, can be regular and systematic in a given period and have a clear enough specification that one can utilize an instrumental theory of heuristic devices and useful rules of thumb in constructing artifactual conversational agents.

In building programs that exhibit the skilled praxis of conversation, we can make progress using a background lore and rules of praxis much as did Gothic cathedral builders employing their practical-technological ingenuity and inventiveness long before there was a science of mechanics. Early in the 18th century, the British Parliament offered a large prize for a solution to the longitude problem since it was difficult for sailors to determine accurately where their ships were located once they lost the sight of land. Galileo, Newton and many physicists had been unable to solve the problem using astronomical theories. A man named John Harrison, with the help of his son, solved it by building a reliable clock, a chronometer, that could keep accurate time while withstanding the buffetings of a sea voyage. Perhaps one day there will be an adequate theory of conversational language. In the meantime, we are building a clock; we are not doing astronomy.

As artisanal engineers, we can proceed instrumentally using our heuristic native-speaker know-how-and-when knowledge of conversational interaction. One advantage of this "pure AI" (purely artifactual) approach is that we are free of the constraints imposed by a theory of how humans process conversational language.

A panelist expressed the view that since every word has a meaning, it may be a mistake to disregard or neglect words in an input as "fluff". But it seems to me that a simplification strategy with special purposes must ferret out and extract from otiose, redundant, superfluous and decoratively cluttered inputs what is important from what is not important for realizing these purposes. This strategy reduces the familiar signal-to-noise ratio characteristic of processing informational messages.

Another panelist sternly proclaimed that it was inhuman to use computers for conversation - a startling moral point to be made this late in the century when it seems clear that people want to talk to computers. We need not take human-human conversation as the gold standard for conversational exchanges. If one had a perfect simulation of a human conversant, then it would be a human-human conversation and not a human-computer conversation with its sometimes odd but pertinent properties. Before there were computers, we could distinguish persons from non-persons on the basis of an ability to participate in conversations. But now we have hybrids operating between persons and non-persons with whom we can talk in ordinary language. Pure machines can only be poked at but these new hybrids are interactive instruments that can be communicated with.

Yorick Wilks disagreed with what he took to be my position that a conversational program need not know anything. I should clarify my point by saying that the program needs knowledge but the question is in what format the knowledge is best represented for the program's purposes. An artifactual conversational modular processor is stipulatively defined as domain specific, informationally encapsulated, special purpose, and having fast, shallow outputs. Thus its knowledge can be efficiently represented in know-how-and-when production rules supplying information about potential action, i.e. what to do in a conversational context. In our module, the know-how rules can generate know-thats, e.g. creating conceptual patterns from the input, and the stored patterns represent know-that knowledge. This position may seem to revive the old and well-chewed procedural-declarative controversy in AI. If so, so be it. In a model of the human mind, declarative prepositional content is perhaps useful. But we are not modeling minds - we are constructing a new type of conversational artifact that can do its job without encyclopedic content.

My panel remarks addressed the problem of why there has been such slow progress in the field of conversational language, and I offered three reasons.

1. Institutional Obstacles - A field competes with rival fads and fashions. In my own experience, it has been very difficult to obtain funding in this area from government sources. I think the ultimate funding will come from the private sector when it realizes how much money can be made from conversing computers.

2. Research Traditions - In growing up in a research tradition our hero or heroine is inspired by, and becomes committed to, its methods, tools and view of the world. Thus in a logic-mathematical tradition, he becomes impressed by the precision of deductive logic, proof theory, the predicate calculus with an equal sign. etc. In a computer science tradition, he is taken with the elegance of Turing machines, recursion, and LISP. In linguistics, he becomes absorbed in parsing, grammars, etc. Then in approaching the problem of conversational language, he tries to use the tools of his research commitment. He underestimates the magnitude of the problem (a very large phase space), and when faced with a tool-to-task misfit, gives up and tries some other problem, preferring to retain the ontological allegiances of his research tradition.

3. Cognitive Laziness - We are all cognitively lazy. We would like to knock off the presenting problem with an X-bar grammar or a theorem-prover. But an adequate conversational language program must have a large semantic data base constructed to a great extent by hand. Machine readable thesauri and even Wordnet must still be added to, subtracted from, and modified by hand to suit the special purposes of the program. There is no escape from large amounts of sheer drudgery and dog-work. ("Hard pounding, this" - Duke of Wellington at Waterloo). This sort of grunt-work is neither magical nor macho programming leading towards a strong AI (somewhere North of the Future) characterized by flourishes of fanciful, unnecessarily complex, and over-elaborate structures so beloved by our hero who wants to show off his programming prowess.

A few "names" appeared in the discussion. I mentioned Chomsky's advice about transformational grammar, namely that it was ridiculous to use one to understand or generate natural language. These are performance phenomena whereas he is talking about competence, an innate language faculty of similar formal operations that delimit the range of natural language grammars. Most languages of the world have no S → NP + VP structure anyway.

It is canonical to refer to Wittgenstein about language but only his name was mentioned. Also missing were Grice's conversational rules (be brief, be clear, be relevant, be truthful).

To Searle's criticism that conversational programs do not really (he means consciously) understand language, David Levy's rejoinder was that for our pragmatic purposes, it makes no difference whether it is a "real" or "simulated" understanding as long as the program delivers the requisite simulated conversational goods.

My son Peter and I were naturally pleased to hear it stated by the panelist Gene Ball from Microsoft that ours was the best conversational program around. Ours represents a coarse-grained strategy at a level of analysis suitable for instrumental purposes in talk exchanges.

For me, the fundamental cognitive concept in all of this is "understanding". We try to understand the way the world works and we try to understand language.

Understanding (sense-making) represents an ontological category relating reality to technology. What we know, we know through the way it makes sense to us. When we write conversational programs, we are engaged in an artisanal technology of building artifactual special-purpose understanding systems. We are at a frontier where no one yet is quite at home. We are faced with insurmountable opportunities (sayeth Pogo) so let's get to it.

3 Human-Computer Conversation in A Cognitive Therapy Program

K.M.Colby

1 Background

I am neither a computational linguist nor an AI linguist nor a linguist of any kind. I am a psychiatrist interested in using computer programs to conduct psychotherapy - traditionally called talk-therapy, harking back to Socrates who remarked "the cure of the soul has to be effected by certain charms - and these charms are fair words".

About 30 years ago I tried to write a psychotherapy program but of course found the natural language problem to be formidable (Colby et al., 1966). The formal grammars and parsers of the time (phrase structure, ATN, etc.) did not give me what I was looking for. Formal operations were simply unable to handle the flux, slop, fluidity, zest and spunk of conversation. Even operating on the immaculate well-formed prose of narrative text, they were too fragile, unwieldy, brittle, slowed by unnecessary complexities, and generally too meticulous to be practical for conducting and sustaining the unruly, tangled, messy, elliptical, motley and cluttered hodgepodge of real-time, real-life conversations. Also these parsers took no account of who was saying what to whom for what purposes. Parsers depended upon a tight conjunction of stringent tests whereas, it seemed to me, what was needed was a loose disjunction of heuristic procedures to make sense out of highly idiolectic expressions characteristic of a talk-therapy context. To get a rough toe-hold on this large phase-space, I wanted a sturdy, robust, pliable, supple, flexible, rough-and-ready, extensible, error-tolerant, sense-making strategy to cut through a sea of noise with a low processing effort in satisfying the task-requirements of therapeutic conversation.

So, with the help of several diligent and courageous graduate students in AI at Stanford (Sylvia Weber Russell, Horace Enea, Lawrence Tesler, William Faught, Roger Parkison) I began to develop a different heuristic sense-making strategy for dealing with the sprawling patchwork melange and morass of idiolectic conversational language (Colby and Enea, 1967; Parkison et al., 1977). Since everyone seems to agree that context, however vague the concept, is important in language and my interests were psychiatric in nature, the psychosocial contexts of our program were those of diagnostic interviewing (as exemplified by interviews with PARRY, a simulation of paranoid thinking (Colby, 1981)) and, for the past few years, cognitive therapy (as currently exemplified by the program Overcoming Depression (Colby, 1995)).

The computer-simulation model PARRY passed several indistinguishability tests in which it was interviewed by expert mental health professionals (psychiatrists, psychologists, psychiatric nurses) who could not distinguish its linguistic behavior from that of paranoid patients at a statistically significant level. PARRY ran nightly on the ARPA network for many years and these interactions, numbering over 100,000 - often just playful or baiting in nature - nonetheless provided us with a rich source of words, phrases, slang, etc. for building a semantic database.

In the 1980s, joined by my son Peter, a highly inventive programmer, I began construction of a conversational Joe-the-Bartender type of program that discussed close interpersonal relationships (Colby et al., 1990). From this program (GURU) we constructed an artifactual cognitive module (domain specific, informationally encapsulated, fast, shallow) to serve as the dialogue mode of a cognitive therapy program for depression entitled Overcoming Depression, which henceforth I will term "the program". With the aid of my wife Maxine and my daughter Erin, we hand-crafted our own large semantic data-base using standard thesauri, several common frequency lists, slang dictionaries, Dear Abby and Ann Landers letters, a corpus of 70,000 words from 35 depressed women who described their life situations, the PARRY data, and the data from many users of the program itself under its years of development. I mention this family-run aspect of the project in passing because all of my several grant proposals to create this program failed to obtain funding from government, private, or university sources. So we founded a little company called Malibu Artifactual Intelligence Works with our own financial resources plus $23,000 from a friend. The program has been commercially available since 1991. Sometimes, when you develop a new field, you have to go it alone without institutional support.

Before getting into the details of the program, I would like to clarify a term I have used several times, i.e. sense-making (Colby et al., 1991). 1 realize this is an excursion into ontology but I will be brief. In person-to-person understanding, we try to make sense of what others say by extracting from their expressions, meaning relevant to our interests in a context. (More on meaning can be found in Electric Words by Wilks et al., (1996)). I take meaning or sense to be a fundamental dimension of reality running through all phenomena. This view is not unique, being shared by physicists or biologists like Wheeler, Bohm and even Crick. Making sense of what other people say is a practice involving extraction of meaningfulness of representational content from the meaning-forness (word-meaning) of language. I will have more to say about this type of sense-making in relation to the ontological status of deponent verbs in the Discussion. Let me now return to the practical perspective our approach to human-computer conversation is embedded in.

2 Computer-Based Cognitive Therapy for Depression

AI is noted for its interest in the workings of the mind. However it is not noted for its interest in one of the mind's main features - mental suffering and its relief. But that is our interest, mainstream or not.

A common example of mental suffering is the cognitive dysfunction of depression which afflicts 25% of the U.S. population at some time or other in their lives. There are roughly three treatments for depression; cognitive therapy, interpersonal therapy, and psychopharmacologic medication. These treatments have been systematically investigated in many controlled clinical trials and have been found to be about equally effective at a rate of 65-75%. The treatments are conducted by human mental health professionals. So why create a computer program? My main reason was to make an end-run around the social-stigma problem which is particularly troublesome in depression in which 65-70% of depressives do not even seek help, mainly because of social stigma. My goal was not to replace live therapists but rather to provide a therapeutic opportunity for depressives where none currently exists. The program does not threaten mental health professionals with unemployment.

Stigma comes from a Greek word meaning to physically brand a person with a mark of infamy or disgrace. A person so marked posed a risk to society, for example an escaped slave, someone not to be trusted. (Even today we hold up our right hand in court to show we are not branded). Although the situation is improving, in our society people with mental-emotional problems are still stigmatized in the workplace and by health insurance companies. Hence my idea was to provide a therapy opportunity in the form of a Personal Computer program that could circumvent social stigma in that the sufferer could engage in a treatment in the privacy and confidentiality of his own home. (Incidentally the method also avoids the negative side effects of some live therapists such as sexual involvements and financial exploitation). To my knowledge, ours is the world's first successful talk therapy program using conversational language. It has been used by thousands of people, attaining a 96% satisfaction rating from a sample of 142 out of 500 users responding to a survey questionnaire. This is by no means a controlled clinical trial but it is observational evidence supportive in the right direction and employing a commonly used global measurement of treatment outcomes. It should not be surprising that our program and chess-playing programs like Deep Blue are so successful. Both involve highly stereotyped situations.

The program is 5 megabytes in size with an interpreter, written by Peter Colby, of about 120K. The program is divided into a text mode and a dialogue mode. The text mode provides concepts and explanations based largely on theories and input sequences. The strategy here is consonant with the natural science strategy of abstraction and simplification by stripping away unessential details. A great range and diversity of manifest patterns of phenomena are pared down by many-to-few transforms and are converted into a much smaller set of stringent underlying patterns in order to make the problem more tractable.

From the set of concordant patterns constructed and fitted, a best-fit pattern is selected as most pertinent and preferred according to a variety of criteria. If there is a tie in the selection process, a random selection of the best-fit is made from the ties constructed. If none of the constructed patterns fit, the program is in a PUNT situation. (In US football, it is a joke that whereas you have four downs, when you don't know what to do on third down, you punt). In this sort of breakdown, the program has two options for recovery. First, it can offer a question or assertion about the topic under discussion or second, it can return to the text mode of the program to the point where it left off to enter the dialogue mode. If he wishes, the user can abort the return to the text and return to the dialogue mode by pressing a special key.

The program has a memory stack for anaphoric reference and for keeping track of topics under discussion. Once a specific LHS pattern is selected, it is time for an associated RHS response to be produced.

3.3 Output Responses

Each LHS pattern of a production rule is linked to a list of 3-10 RHS responses. These responses are ordinary language expressions of three syntactic and five semantic types. The syntactic response types are Questions, Requests, and Assertions. The semantic response types refer to Beliefs, Causes, Feelings, Desires, and Actions.

The RHS responses are partially ordered in respect to semantic intensity from milder to stronger as the dialogue proceeds. By the term 'stronger" here we mean the output expression points more directly to the user's personal intentional system.

Each RHS response is made up of a formula of constants and variables with a list of options for the variable placeholders. For example, in the assertion "You feel your wife nags you", the word "feel" has been randomly selected from the list "feel", "think", "believe", the word "wife" from the list "wife", "spouse", "mate", and the word "nags" from "nags", "berates", "criticizes". These lists are roughly synonymic but sometimes contain a contrast class or a "goes-with" expression. These constant-variable formulae allow the generation of a great variety of output surface expressions with holding power, an ability to keep the conversation going. Currently the program can generate about 600,000 responses in this way.

The meaningful content of the responses stem from our own psychologic inferences. The program does not reason from premise to conclusion on its own. To repeat, it is a specialized artifactual cognitive module, domain specific, informationally encapsulated, fast and shallow. It is a psychologic inference engine grinding out its authors' inferences. It succeeds partly because of its designed ignorance. The activity- specific knowledge structures are appropriate to their use.

4 Discussion

Numerous questions have been raised about our modular strategy. For example, does this module "understand" natural language? Much depends on who is asking the question and what one means by "understand". Often the issues are only terminological. If Searle, of Chinese Room fame, is asking the question, I would answer "No - it does not understand in your sense. It only behaves as if it understands, which is good enough to get the job done". Our modular processor can carry on conversations without having explicit propositional content-representations of what is being talked about.

To someone else, I might answer that the module has a type of restricted know-how understanding in that it knows how to react to the meaning-fors, the convention-governed word-meanings, of linguistic input but does not consult propositional content-carrying meaningful states of a representational system. The modular processor uses "she" as an anaphoric reference to wife but it does not contain the propositional knowledge that a wife is a woman. This know-that knowledge is only implicit in the control structures.

How does the module know what to do? We have designed it to do what it does, e.g. form patterns, fit them, and momically respond with the RHS of a production rule. It has skill in dealing with words properly in this context. Since its know-how knowledge of what to do is installed implicitly in condition-action production rules, its knowledge is not available for more general use, for example, by a central inference processor with encyclopedic knowledge or by other modules.

Is the dialogue mode a simulation of a human cognitive therapist? No - it is not intended to represent a simulation or imitation of a human therapist. At times the responses resemble those of a human but that is only because the program's authors simulate themselves in designing cogent responses, i. e. responses consistent with the interpretation that the program has a therapeutic intent. Recall my mention of the virtual person in the dialogue mode. This conversational participant says many things a human therapist would never say, e.g. "I am sure my programmers would be glad to hear that" in response to a user compliment. Who is this "I" and "my"? It is a conversational participant with a particular character and set of attitudes that we have constructed. One might view its presence as a type of theater, thus lending the flavor of an art-form to the program.

Is our modular artifact intelligent? Yes and no. It selects the right thing to do at the right time under the circumstances and under time constraints. But it obviously is not a complete or general intelligent agent.

Early on I alluded to an ontological point about sense-making and deponent verbs. These verbs, such as "think", derive from Latin passives with an active sense. Do I think or is there something in me that thinks? Nowadays we seldom use the archaic "methinks" but we still say "it occurred to me".

What is agent and what is patient? Is thinking something I do or something that happens to me? Or both? What agent deserves the deictic "I"? Does our module think, or are we building only a talker rather than a thinker? Does it make sense, or

Illinois, Open Court Press.

Wilks, Y.A., Slator, B.M. and Guthrie, L.M. 1996. Electric Words: Dictionaries, Computers and Meanings. MIT Press, Cambridge, MA.

4 Architectural Considerations for Conversational Systems

G. Görz, J. Spilker, V. Strom and H. Weber

1 Conversational Requirements for Verbmobil

Verbmobil[1] is a large German joint research project in the area spontaneous speech-to-speech translation systems which is sponsored by the German Federal Ministry for Research and Education. In its first phase (1992-1996) ca. 30 research groups in universities, research institutes and industry were involved, and it entered its second phase in January 1997. The overall goal is develop a system which supports face-to-face negotiation dialogues about the scheduling of meetings as its first domain, which will be enlarged to more general scenarios during the second project phase. For the dialogue situation it is assumed that two speakers with different mother tongues (German and Japanese) have some common knowledge of English. Whenever a speaker's knowledge of English is not sufficient, the Verbmobil system will serve him as a speech translation device to which he can talk in his native language.

So, Verbmobil is a system providing assistance in conversations as opposed to fully automatic conversational systems. Of course, it can be used to translate complete dialogue turns. Both types of conversational systems share a lot of common goals, in particular utterance understanding - at least as much as is required to produce a satisfactory translation, processing of spontaneous speech phenomena, speech generation, and robustness in general. A difference can be seen in the fact that an autonomous conversational system needs also a powerful problem solving component for the domain of discourse, whereas for a translation system the amount of domain knowledge is limited by the purpose of translation, where most of the domain specific problem solving - except tasks like calendrical computations - has to be done by the dialog partners.

To enable component interaction, we designed the communication framework ICE (Amtrup, 1995; Amtrup and Benra, 1996) which maps an abstract channel model onto interprocess communication. Its software basis is PVM (Parallel Virtual Machine), supporting heterogeneous locally or globally distributed applications. The actual version of ICE runs on four hardware platforms and five operating systems with interfaces to eight programming languages or dialects.

4 Interactions between Recognizer, SynParser, SemParser, and Prosody

To understand the operation of INTARC, we start with an overview of its syntactic parser component (SynParser). Whereas the dialogue turn based grammar of the system is a full unification grammar written in HPSG, SynParser uses only the (probabilistically trained) context-free backbone of the unification grammar - which overgenerates - *and* a context-sensitive probabilistic model of the original grammar's derivations. In particular, the following preprocessing steps had to be executed:

1. Parse a corpus with the original unification grammar G to produce an ambiguous tree bank B.
2. Build a stripped-down (type skeleton) grammar G' such that for every rule r' in G' there is a corresponding rule r in G and vice versa.
3. Use an unsupervised reestimation procedure to train G' on B (context sensitive statistics).

The syntactic parser (SynParser) is basically an incremental probabilistic search engine based on (Weber, Spiker and Görz, 1997) (for earlier versions cf. Weber, 1994; Weber, 1995)), it receives word hypotheses and phrase boundary hypotheses as input. The input is represented as a chart where frames correspond to chart vertices and word hypotheses are edges which map to pairs of vertices. Word boundary hypotheses (WBHs) are mapped to connected sequences of vertices which lie inside the time interval in which the WBH has been located. The search engine tries to build up trees according to a probabilistic context free grammar supplied with higher order Markov probabilities. Partial tree hypotheses are uniformly represented as chart edges. The search for the n best output trees consists of successively combining pairs of edges to new edges guided by an overall beam search strategy. The overall score of a candidate edge pair is a linear combination of three factors which we call decoder factor, grammar factor and prosody factor. The decoder factor is the well-known product of the acoustic and bigram scores of the sequences of word hypotheses covered by the two connected edges. The grammar factor is the normalized grammar model probability of creating a certain new analysis edge given the two input edges. The prosody factor (see next section) is calculated from the acoustic WBH scores and a class based tetragram which models sequences of words and phrase boundaries.

As mentioned in the last section, one of the main benefits of prosody in the INTARC system is the use of prosodic phrase boundaries inside the word lattice search.

When calculating a prosody factor for an edge pair, we pick the WBH associated with the connecting vertex of the edges. This WBH forms a sequence of WBHs and word hypotheses if combined with the portions already spanned by the pair of edges. Tests for the contribution of the prosody factor to the overall search lead to the following results: For a testset with relative simple semantic structure the use of the detected phrase boundaries increased the word recognition rate[2] from 84% to 86% and reduced the number of edge pairs (as a measure for the run time) by 40%. For the 'harder' Verbmobil dialogues prosody raised the word recognition rate from 48.2% to 53.2% leaving the number of edge pairs unchanged.

In INTARC, the transfer module performs a dialog act based translation. In a traditional deep analysis it gets its input (dialog act and feature structure) from the semantic evaluation module. In an additional path a flat transfer is performed with the best word chain from the word recognition module and with focus information.

During shallow processing the focus accents are aligned to words. If a focus is on a content word a probabilistically selected dialog act is chosen. This dialog act is then expanded to a translation enriched with possible information from the word chain.

Flat transfer is only used when deep analysis fails. First results show that the 'focus-driven' transfer produces correct - but sometimes reduced - results for about 50% of the data. For the other half of the utterances information is not sufficient to get a translation; only 5% of the translations are absolutely wrong.

While the deep analysis uses prosody to reduce search space and disambiguate in cases of multiple analyses, the 'shallow focus based translation' can be viewed as directly driven by prosody.

5.2 Speaker Style

A new issue in Verbmobil's second phase are investigations on speaker style. It is well known that system performance depends on the perplexity of the language models involved. Consequently, one of the main problems is to reduce the perplexity of the models in question. The common way to approach this problem is to specialize the models by additional knowledge about contexts. The traditional n-gram model uses a collection of conditional distributions instead of one single probability distribution. Normally, a fixed length context of immediately preceding words is used. Since the length of the word contexts is bound by data and computational resources, practicable models could only be achieved by restricting the application domain of a system. Commonly used n-gram models define $P(w|C,D)$ where C is a context of preceding words and D is an application domain. But also finer grained restrictions have been tested in the last decade, e.g. a cache-based n-gram (Kuhn and DeMori, 1990).

Intuitively, every speaker has its own individual speaking style. The question is whether it is possible to take advantage of this fact. The first step towards