Clinical Decision Support for the Information Age: Evaluating the Role of Expert Systems in the New Medical Information Paradigm

Permalink
https://escholarship.org/uc/item/0gb429z1

Author
Birdsong IV, William H

Publication Date
1995-04-01

License
CC BY-NC-ND 4.0
CLINICAL DECISION SUPPORT FOR THE INFORMATION AGE

Evaluating the Role of Expert Systems in the New Medical Information Paradigm

by

William Herbert Birdsong

B.A. (University of California, Santa Cruz) 1988

A thesis submitted in partial satisfaction of the requirements for the degree of

Master of Science

in

Health and Medical Science

in the

GRADUATE DIVISION

of the

UNIVERSITY of CALIFORNIA at BERKELEY

Committee in charge:

Assistant Professor Mark J. Van der Laan, Ph.D., Chair
Professor Jerome A. Feldman, Ph.D.
Associate Professor Marcus Thygeson, M.D.

1995
The thesis of William Herbert Birdsong IV is approved:

Chair

Date

Date

Date

University of California at Berkeley

1995
# Table of Contents

Acknowledgments ........................................................................................................ iv
Introduction .................................................................................................................. 1

Origins of Expert Systems: Artificial Intelligence ......................................................... 2
  Experimenting with Artificial Intelligence ................................................................. 6
  General Problem Solver ....................................................................................... 10

Designing Medical Expert Systems ............................................................................ 16
  Representing Knowledge in Expert Systems ......................................................... 19
  Propositional Logic ............................................................................................... 20
  First-Order Predicate Calculus ............................................................................ 23
  Medical Reasoning is Probabilistic ...................................................................... 23
  Sensitivity and Specificity ................................................................................... 25
  Bayes' Theorem and Posttest Probability .............................................................. 26
  Constructing a Knowledge Base ......................................................................... 27


Quick Medical Reference and Iliad: State of the Art Systems ................................. 33
  Internist & Quick Medical Reference ................................................................. 33
  The Iliad Clinical Consulting System ................................................................ 38

Conclusions ............................................................................................................... 46
  Why Use Expert Systems? .................................................................................. 46
  Are We There Yet? ............................................................................................. 49
  Closing the Information Loop ............................................................................ 52

Attitudes Toward Medical Information Technology ............................................... 56

Computer Literacy and Education ....................................................................... 60

Legal and Ethical Considerations .......................................................................... 61

Politics, Power, and the Art of Electronic Medicine .............................................. 67

The Corporate Revolution .................................................................................... 71

The IT Medium and the Message of Managed Care .............................................. 72

References ............................................................................................................. 76
ACKNOWLEDGMENTS

My thanks goes out to all of those who have guided me toward a better understanding of medical informatics and my future role in the medical profession. To Marcus Thygeson, M.D. for inspiring me to look into the subject of medical informatics and for modeling the attributes of a great physician. To Mark Van der Laan, Ph.D. for encouraging me and for putting me at ease. To Jerome Feldman, Ph.D. for making me remember that this is a master's degree in science and for giving me a way of finishing it. To Henrik Blum, M.D. for encouraging me to think about the broader role of medical information technology. To Ronnie London, for coordinating a soft landing. To Steve Yao for convincing me to "wrap it up." To Jane Givens for her friendship. And to my wife, Bobbie for believing in me.
Introduction

In the February 23, 1990, issue of JAMA [1, 2], Robert Greenes and Edward Shortliffe defined informatics as

"... the field that concerns itself with the cognitive, information processing, and communication tasks of medical practice, education, and research, including the information science and the technology to support these tasks. An intrinsically interdisciplinary field, medical informatics has a highly applied focus, but also addresses a number of fundamental research problems as well as planning and policy issues. Medical informatics is now emerging as a distinct academic entity. Healthcare institutions are considering, and a few are making, large-scale commitments to information systems and services that will affect every aspect of their organizations' function."

They add that the emergence of medical informatics is due to:

... advances in computing and communications technology, to an increasing awareness that the knowledge base of medicine is essentially unmanageable by traditional paper-based methods, and to a growing conviction that the process of informed decision making is as important to modern biomedicine as is the collection of facts on which clinical decisions or research plans are based.

Expert systems, or knowledge-based systems, are computer programs that analyze data in a way that, if performed by a human, would be considered intelligent. Expert systems are characterized by: (1) symbolic logic rather than just numerical calculations, (2) an explicit knowledge base that is understandable to an expert in that area of knowledge, and (3) the ability to explain conclusions with concepts that are meaningful to the user. Expert systems can be useful in two different ways: (1) Decision support - to remind an experienced decision maker of options or issues to consider that he or she once knew but may have forgotten. This is the most common use in medicine. (2) Decision making - to allow an unqualified person to make a decision that is beyond his or her level of training and experience. The first medical expert system was developed in the early 1970's by Dr. Edward Shortliffe at Stanford University. It recommended the selection of antibiotics based on clinical data such as the site of infection and associated medical conditions. While Shortliffe's was not the first decision support program, it was the first to use symbolic knowledge in a rule-based format. Over the past two decades, the computer programming methods used to create expert systems ("knowledge engineering") have been incorporated into the standard software engineering repertoire of techniques.
This thesis is on the role of expert systems in the emerging medical information paradigm. My aim is to place expert systems in their historical context, develop a basic understanding of their technical basis, analyze the current state of the art, and discuss the issues raised by their implementation within the broader context of managed care and the coming medical information infrastructure.

While reading this thesis, keep in mind the following questions: What will medicine be like in fifty years? Will physicians and nurses speak highly of the ways in which computers have been integrated into medical practice? Will medicine remain an art? Will medicine seem more interesting and intellectually stimulating and offer better opportunities to learn and assimilate new material? Will practitioners become more efficient and use their extra time for scholarship and patient interaction? Might the practice of medicine even become less stressful? Most importantly, will patients have better access to a higher quality of medical care and live healthier lives as a result of the products of medical informatics? My hope is that by studying expert systems, we may gain insight into where medicine is currently headed and the ways in which we will be challenged by new developments in medical information technology.

 Origins of Expert Systems: Artificial Intelligence

Expert systems are one of many results of artificial intelligence research. In this section, I intend to explain by example what artificial intelligence is, a little bit about its methodology, and how its early failures ultimately led to a new approach in expert systems. My intention is not to develop a highly technical explanation of AI methodology. Rather, I will refer to just a few of the field's definitive projects and elaborate with examples of my own. My purpose is to establish a basis for understanding how expert systems evolved out of AI research and to establish a context for understanding the issues involved in designing and evaluating them in the medical domain.

Artificial intelligence is a science that captures the imagination like few others, and whose name "... combining as it does a highly immodest ambition with a suggestion of deceit ... has the power to provoke controversy [3]." We are at once fascinated and frightened by its dialectical implication: that we may be clever enough to create a machine in our own image and by that act, establishing once and for all that being clever is no more of a claim to divine
preference than being ferocious or fast. The very idea of artificial intelligence can be threatening, especially to those who gain or succeed by virtue of possessing the natural variety. Therefore, to win confidence in AI, I will attempt to present it tactfully, as did Marvin Minsky who described it as "the science of making machines do things that would require intelligence if done by men [2]." With this definition, we might be tempted to sidestep entirely the philosophical question of whether or not machines could ever really possess intelligence and focus on more practical issues like how we might merely make them seem to. After all, the theoretical possibility of machine intelligence is a controversial subject and our primary goal is to understand the basis for medical expert systems. However, the issue turns out to be relevant in a historical context since the explosion of expert systems research in the 1970s is rooted, in part, in a crisis of faith in the theoretical possibility of machine intelligence as defined at the outset of the AI movement. Therefore, let us focus briefly on this definition in order to fully appreciate how expert systems developed out of AI.

The classical distinction between artificial intelligence and mere automation was proposed by Alan Turing, a British mathematician who designed (among other things) an electromechanical computer used to crack German codes during World War II. Turing is remembered best for proposing a test (the Turing test) that would establish whether or not a computer possessed intelligence. In this test, a man would converse via a terminal with two hidden subjects, a woman and a computer programmed to act like a woman. If, after a period of interrogation, the man could not guess which subject was which, then the computer passed the test [4]. Turing may have been influenced by his contemporary, the psychologist B.F. Skinner, who proposed that internal mental processes (in laboratory rats as well as human beings) were irrelevant since what really mattered was behavior. The proper focus of psychologists, he asserted, was the relationship between stimulus and response. In a similar manner, Turing focused computer scientists' attention on the relationship between input and output. It didn't matter to him how output was derived as

1 Physicians have historically claimed membership in this class no more out of conceit than consensus.
long as it demonstrated the qualities he desired, namely, the gift of gab and an
inclination toward mendacity or female impersonation.

Taking Turing’s argument to its logical limit, computer intelligence can be seen
to span a spectrum much like the human variety. There are idiot savants (like
the unintelligent but once astonishing mechanical adding machines) and there
are geniuses (like the fictional but intriguing HAL-9000 of Stanley Kubrick’s film
2001: A Space Odyssey). Our standard depends on our expectations. When I
refer to the theoretical possibility of machine intelligence in this chapter, I
generally mean the possibility that machines could ever fall on the “Kubrick
side” of a line dividing this spectrum into intelligent and unintelligent entities
based on an arbitrary standard such as Turing’s and our critical assessment of
whether that standard is met by the observed relationships between the
machine’s input and output. In general, our standard will remain in the range
suggested by Turing and differ only on the basis of the specific task we have in
mind. Thus, a machine that can produce meaningful differential diagnoses
based on various signs and symptoms seen in a real patient and suggests ways
of gaining certainty in a smaller subset of that differential is by definition an
artificially intelligent medical computer.

Notice that I retain the designation “artificial.” This is to remind myself and
others that actual intelligence is probably beyond the capability of a machine
that operates by implementing a program, as most still do. The fact is, this is not
how humans think. If we are to adhere to the strictest standard of human
intelligence, then we must accept that it goes far beyond the production of
meaningful output and encompasses such domains as emotion, feelings, the
subconscious, mood, and many very complicated processes that are regulated
biochemically, hormonally, and with subtlety that continues to boggle some of
the greatest minds in science.

Those who believe otherwise are proponents of strong AI as defined (and
refuted) by John Searle of U.C. Berkeley [5]. Proponents of strong AI believe
that a machine could think by implementing a computer program. The basis for
this belief is the hypothesis that thought itself is primarily the ability to
manipulate symbols (according to a set of instructions in most cases) in order to
produce meaningful output. Searle refutes this position by pointing out how a
machine such as Turing’s merely simulates intelligence without actually
replicating it. To Searle, thought is defined by its biological basis. Any machine that can mimic the output of intelligent humans is by definition no more than artificially intelligent.

"Barring miracles, you could not run your car by doing a computer simulation of the oxidation of gasoline, and you could not digest pizza by running the program that simulates such digestion. It seems obvious that a simulation of cognition will similarly not reproduce the effects of the neurobiology of cognition.”

Searle's point is that true intelligence is defined by neurobiology and not by meaningful output. This is not to say that machine intelligence is useless. In fact, the programs that we will review later in this chapter are rather impressive despite the validity of the argument that they do not actually think. Therefore, in this paper, we will take the position of weak AI. That is, we will acknowledge the value of AI methods while leaving the question of whether machines will ever truly think to philosophers and, perhaps, our great grandchildren.¹

AI actually started out as an ambitious effort to create machines that exhibited what might be called "general intelligence". The premise was that this could be accomplished by dividing rational behavior into two parts, symbolic representations of reality (data) and a process for manipulating them. The results of this process would be the generation of new content which could either be output (if, for example, the input had consisted of a query) or used to accomplish a goal (if the input consisted of a command). Investigators reasoned that once such a process had been perfected and codified, the limits of machine intelligence would depend only on the amount of data one made available. As ambitious as this may sound, early investigators were not only hopeful, they were confident! Turing, for example, was certain that a computer would pass his test by the year 2000. Others expressed their enthusiasm in more bombastic terms:

It is not my aim to surprise or shock you- but the simplest way I can summarize is to say that there are now in the world machines that think, that learn and that create. Moreover, their ability to do these things is going to increase rapidly until in a visible future- the range of problems they can handle will be coextensive with the range to which the human mind has been applied. [6]

¹Searle believes that this may very well be possible but that it will require far more than the implementation of a program.
How did these gentlemen come to possess so much confidence in a field that was less than a decade old? Consider their early success:

Experimenting with Artificial Intelligence

In the early days of AI, there was no clear distinction between cognitive science (now a separate but related discipline that seeks to model human intelligence) and the development of practical tools. The Turing Test established a human-like standard for the behavior of AI and many considered the most promising process of manipulating symbols to be some version of what humans do when they think. To them, modeling human intelligence and creating practical applications were really the same thing. Therefore, the problem was how to formalize and encode the process of rational (human) thought.

When presented with a problem, it is often a good idea to consider whether a solution already exists before thinking one up from scratch. Herbert Simon (of Carnegie Tech, now Carnegie Mellon) and Allen Newell (of the RAND Corporation) considered the problem of formalizing and encoding the process of rational thought and immediately realized that a solution was available, and from a source no less hallowed than Aristotle! Logic is an internally consistent, well described formalism for solving problems via the manipulation of symbols. Might its principles be incorporated into an algorithm that could then be run by a machine to solve real problems? If so, how well would it perform?

To answer these questions, Newell and Simon began work on a new kind of program. The idea was to produce an effective algorithm for applying the rules of logical inference to logical propositions in proper sequence to solve theorems in the domain of geometry. In short, to replicate the performance of a logician.

When a logician approaches a proof, he must decide which inference rule to apply and where. He gains this knowledge through experience. As he evolves from novice to expert, he comes to understand what works and what doesn't. Each proof may be unique, but over time he learns to see how they are the same. If we were looking over his shoulder, we might ask him how he had come to choose a particular maneuver. If he responded by saying that he had been in similar situations before and had learned a few tricks for progressing
out of them, then we would say that he had used a *heuristic*, which is a rule of thumb or a partial explanation based on incomplete evidence. On the other hand, he might simply shrug and say that it was a hunch, in which case we would call it *intuition*. In either case, we would say that he understood the problem.

Now, regardless of how we feel about computers and whether they will ever come to possess anything like true intelligence, we know that in 1955, none were capable of actually understanding anything. How then might a mass of transistors (integrated circuits were still three years off) be made to solve a complex geometric proof? The answer is somewhat discouraging, for it leaves us with an even more profound understanding of the "artificial" nature of machine intelligence. Newell and Simon proposed that while they clearly couldn't program a computer to understand geometry, they might very well succeed in programming one to solve such geometric problems by brute force! That is, like a high school sophomore who failed to study for an exam, a computer might randomly, or better yet, systematically apply each inference rule to various sets of propositions until eventually, the proof is solved. Given a small set of propositions and a simple proof, a student might actually succeed in this manner\(^1\). But with larger proofs, things would become more complicated and would quickly get out of hand as the number of possible combinations of manipulations grew exponentially with each step.

But to a computer, this isn't a problem at all. With a moderate amount of memory and the ability to rapidly read symbols, transform them, and write the results back to memory, the search for the right combination of manipulations becomes a more or less trivial matter of bookkeeping. A computer could systematically apply the rules of logical inference to all possible subsets of a set of axioms to generate a new set of propositions. The union of the original set of axioms plus the new set of propositions would then serve as input for another round of inference which would create still more propositions and so on. By

\[\]

\(^1\) I have to pause to wonder how Alan Turing would grade the poor student's test after inspecting the ridiculous combination of steps he had taken to arrive at the correct answer. My intuition is that the Turing Test imposes a double standard. The student would get an F, the computer, an A+!
iteratively applying the rules of logical inference to an expanding list of true statements, Newell and Simon reasoned that *Logic Theorist* would eventually generate the proposition it sought to prove. The task was therefore to come up with a systematic way of choosing combinations of rules and axioms and a way of mapping one symbolic representation of a proposition to another based on the rule being applied.

There was only one problem. In order to implement their idea on a computer, Newell and Simon had to invent a whole new programming language. FORTRAN, the best thing going at that time, had serious shortcomings when it came to implementing their theorem prover concept because it required that all variables and the relations between them be declared *before* any processing could begin. This worked well for most engineering and scientific applications, but it was ill suited for theorem proving which would require the ability to constantly create or destroy new symbols on the fly. After simulating their approach by hand (passing various notecards between family members and graduate students) they convinced themselves that it would work and decided that creating such a language would be worth the effort.

Within about one year, they had their new language, IPL, the first to incorporate a "list-processing" technique which would eventually be incorporated by John McCarthy into LISP (List Processing Language) a language that is still used for AI applications today\(^1\). Newell and Simon wrote the code for their theorem prover and tested it out on some of the more difficult proofs found in Russell and Whitehead's *Principia Mathematica*. To their delight, *Logic Theorist* succeeded in proving thirty-eight of the first fifty-two theorems in chapter two of the *Principia*.\(^2\)

Newell and Simon presented *Logic Theorist* in 1956 at the Dartmouth Summer Seminar, which is often cited as being the first AI conference ever convened. Daniel Crevier’s account of this conference [2]. suggests that Newell and Simon’s presentation of an actual working AI program was more infuriating than

---

\(^1\) Because of its peculiar syntax, computer science students often assert that LISP stands for "Lots of Infuriating Superfluous Parentheses."

---

8
inspiring to many of those in attendance. Most had come hoping to gain enough knowledge to return to their labs and win the race to create the first AI program. Therefore, they weren’t pleased to find that the race was already over. The fact that the victors were somewhat immodest (by their own account [2]) certainly didn’t help. With Cartesian humility, Simon would later write [6] that they had:

... invented a computer program capable of thinking non-numerically, and thereby solved the venerable mind/body problem, explaining how a system composed of matter can have the properties of mind.

The Dartmouth Seminar is cited as the beginning of the AI movement for it was there that the community took shape and certain basic principles were first considered. Among those principles was the assertion that:

Every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it [6].

This statement came to be known as the "physical symbol system hypothesis." It is an assertion that intelligence is the ability to manipulate symbolic representations of the world. Those who believe this hypothesis to be true may reason by implication that a machine endowed with sufficient capacity for symbolic manipulation would theoretically be capable of possessing true intelligence. In fact, one could even argue that if the hypothesis is true, then the distinction between real and artificial intelligence is actually false and that any object capable of manipulating symbolic representations of the world is intelligent - to the degree that those manipulations were effective. Strong AI was off to an enthusiastic start.

Which leads back to the idea of a spectrum wherein the important factor is not so much how, but how well the object performs and at what tasks. Newell and Simon showed that a computer could complete a mathematical proof. But Turing would have it engage in polite conversation. In the decade after the Dartmouth Conference, anything seemed possible. If intelligence was nothing more than the ability to manipulate symbolic representations of the world, then

---

2 One of these proofs was actually more elegant than the one proposed by the authors!
general intelligence seemed only to require improving that ability and expanding the domain of symbols to which it could be applied.

General Problem Solver

Over the course of the next decade, AI research flourished at institutions like Carnegie Mellon, MIT, Stanford and IBM. Newell and Simon moved on to a new project they hoped might demonstrate general intelligence by more accurately simulating human problem solving. Recent psychological investigations had shown that when solving problems, people don't actually reason like Logical Theorist. Rather, they use a general set of heuristics that we might call common sense to reduce the difference between the way things are and the way they would like them to be. That is, when humans solve problems, they rarely think in terms of logical inference as in - "I am hungry therefore I should seek food." The process is more like- "I am hungry. How can I make that hunger go away?" The answer depends on what is available in the environment (a refrigerator or a telephone, for example) and what is known about those things (refrigerators contain food, phones can be used to order pizza). Newell and Simon incorporated this perspective within a new program, General Problem Solver (GPS), analyzed such mundane knowledge about objects and actions to construct stepwise solutions to tasks through a process called means-ends analysis.

Initially, the following types of information were made available to GPS: (1) An initial state with a number of attributes representing the locations and conditions of an agent (a fictional character that can be thought of as a robot or video game character), and the various objects in its environment. (2) A goal state with attributes representing the desired locations and conditions of the agent and the objects in its environment. (3) A repertoire of actions that the agent is able to perform along with their preconditions and consequences.

Table 1 depicts the initial and goal states for a hypothetical (and somewhat simplified) implementation of GPS involving a "hungry agent." The problem that GPS was programmed to solve was to find a sequence of actions that would move the agent from its initial state through a number of transition states to the goal state. The strategy was to identify differences between the initial and goal states and eliminate them by performing an action known (via access to
something like table 2) to reduce that difference by the addition or subtraction of one or more state attributes.

Table 1. Initial and Goal States

<table>
<thead>
<tr>
<th>Initial State</th>
<th>Goal State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent at home</td>
<td>Agent at home</td>
</tr>
<tr>
<td>Agent is hungry</td>
<td>Agent is happy</td>
</tr>
<tr>
<td>Agent in chair</td>
<td>Agent in chair</td>
</tr>
<tr>
<td>Money in drawer</td>
<td>Money in drawer</td>
</tr>
<tr>
<td>Drawer is closed</td>
<td>Drawer is closed</td>
</tr>
<tr>
<td>Scout at door</td>
<td>Scout at door</td>
</tr>
<tr>
<td>Scout has cookies</td>
<td>Scout has cookies</td>
</tr>
</tbody>
</table>

In our example (table 1), you can see that the goal state lacks one attribute present in the initial state, "Agent is hungry." On detecting this difference, GPS searches through the agent's repertoire for an action that appropriately subtracts "Agent is hungry" from its state (table 2) and finds that there is one action, "Eat cookies," that accomplishes this. But eating cookies has a precondition. The agent's state must include the attribute "Agent has cookies."
<table>
<thead>
<tr>
<th>Action</th>
<th>Preconditions</th>
<th>Remove</th>
<th>Add</th>
</tr>
</thead>
<tbody>
<tr>
<td>Go to desk</td>
<td>Agent at home</td>
<td>Agent at door</td>
<td>Agent at desk</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Agent in chair</td>
<td></td>
</tr>
<tr>
<td>Open drawer</td>
<td>Agent at desk</td>
<td>Drawer closed</td>
<td>Drawer open</td>
</tr>
<tr>
<td></td>
<td>Drawer closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Get money</td>
<td>Agent at desk</td>
<td>Nothing&lt;sup&gt;1&lt;/sup&gt;</td>
<td>Agent has money</td>
</tr>
<tr>
<td></td>
<td>Drawer open</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Money in drawer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Go to door</td>
<td>Agent at home</td>
<td>Agent at desk</td>
<td>Agent at door</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Agent in chair</td>
<td></td>
</tr>
<tr>
<td>Pay scout</td>
<td>Agent has money</td>
<td>Agent has money</td>
<td>Scout is paid</td>
</tr>
<tr>
<td></td>
<td>Agent at door</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Scout at door</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Get cookies</td>
<td>Scout at door</td>
<td>Nothing&lt;sup&gt;2&lt;/sup&gt;</td>
<td>Agent has cookies</td>
</tr>
<tr>
<td></td>
<td>Scout has cookies</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Agent at door</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Scout is paid</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sit in chair</td>
<td>Agent at home</td>
<td>Agent at door</td>
<td>Agent in chair</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Agent at desk</td>
<td></td>
</tr>
<tr>
<td>Eat cookies</td>
<td>Agent has cookies</td>
<td>Agent is hungry</td>
<td>Agent is happy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cookies are solid</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Agent has cookies</td>
<td></td>
</tr>
</tbody>
</table>

<sup>1</sup>We assume that the drawer contains plenty of money.

<sup>2</sup>We assume that the scout has lots of cookies.
Does the agent have cookies? GPS searches its state and concludes that it does not. At this point GPS could either give up on having the agent eat cookies and search for a different action that would eliminate the hunger attribute from its state (in this case there isn't one) or, search for an action that would remedy the lack of cookies.¹

It turns out that cookies can be attained by performing the action "Get cookies." However, before having the agent get cookies and thereby make a commitment to eating them, GPS needs to make sure that eating cookies does not also eliminate or create some other attribute that should or should not be in its goal state (like "Agent is fat"). A quick search reveals that two other attributes are removed from the agent's state by performing the "Eat cookies" action, "Agent has cookies" and "Cookies are solid." (Apparently the cookies are fat free). Since neither of these attributes are desired in the goal state, GPS concludes that "Eat cookies" is a safe action to perform and proceeds to analyze its means of getting some.

But "Get cookies" also has preconditions. Before performing this action, the agent's state must possess several attributes. There has to be a scout at the door, the scout has to have cookies, the scout has to be paid, and the agent has to be at the door. This time there are two attributes that are lacking, "Scout paid", and "Agent at door." Now things start to get complicated. GPS has to make some choices. Again, one possibility is to give up on getting cookies as a way of having them and to search for another way (such as by baking them which our particular agent cannot do). Another possibility is to define each of the unmet preconditions as subgoals and proceed in the same manner as has already been described to try to find sequences of actions to satisfy their preconditions so that the primary goal state can be reached. The algorithm terminates when either the goal state is reached (GPS succeeds) or after exploring all branches of the search tree, no solution can be found (GPS fails).

¹Whether GPS move on to another action or explores the current one first depends on how it is programmed. Which strategy is more efficient depends on the parameters of the "search space", the likelihood of finding a solution at any particular depth.
You might infer, therefore, that the power of GPS is determined by the size of its repertoire of actions and by the granularity of its knowledge about the consequences of actions. For example, our "hungry agent" appears capable of occupying only three positions within its home. It can be at the door, at the desk, or in the chair. Therefore, any problem requiring another position is unsolvable unless a new action is added that would get the agent to the required place. From a practical point of view this may not seem like a problem. The addition of problem specific knowledge is relatively trivial and merely requires an understanding of the kinds of problems that you would like GPS to solve. But therein lies a fundamental difficulty. How can a program demonstrate intelligence if it is only capable of solving problems that had already been foreseen?

To demonstrate, imagine a modified Turing test for an extremely knowledgeable GPS-like program. As in the Turing test, a witness must decide which agent is human and which is controlled by a computer. However, in this test, instead of conversing with the agents, the witness gives them commands. To make things fair, the agents are placed in identical rooms containing identical objects and the witness is instructed to give only commands that can be accomplished within the room using the objects found there. In anticipation of this test, we have given our GPS-like program an immense repertoire of actions. Our program knows about every object in the room and the preconditions and consequences of every action that can be performed with them. How confident can we be that our program will pass the test?

The witness begins by telling the agents to change the light bulb. This might be enough to undo some programs, but ours knows that the bulb provides the only light in the room, that there is a flashlight in the closet, that there are batteries in the drawer, and that the agent can use a chair to reach the bulb. Our program might even gain an early advantage by knowing to wait until the bulb cools before unscrewing it while the human agent burns his fingers. Then, in problem after problem, the GPS-driven agent performs as well as or better than the human. We are pleased as our efforts to account for everything are rewarded. Nothing outsmarts our agent as it even knows how to do absurd things like balance a broom and stand on its head. Then, just before time is called, the witness tells the agents to throw the light bulb across the room. Having thought
of "throw", we were smart enough to include it in our agent's repertoire. But we forgot that there was one object in the room that could easily be broken. Both agents throw their light bulbs causing them to shatter into thousands of pieces. The witness then tells the agents to sweep up the pieces. The human agent obeys, and sweeps the fragments into a neat pile. But our agent is sunk. The room is now filled with thousands of objects our agent cannot handle. Moreover, it is missing one light bulb. To add insult to injury, the witness proceeds to tell both agents to exchange the light bulbs again. The human agent correctly fails to perform this action. But our agent, being in a universe it could not anticipate, begins searching the room, endlessly looking for an object that no longer exists. If you think that we might have anticipated this and programmed our system accordingly, consider the fact that on rare occasions, the light bulb would not break.

This example demonstrates some of the arguments that were leveled against AI by critics in the late 1960s and early 1970s. In 1969 McCarthy and Hayes addressed some of the philosophical problems posed by AI [7]. One of their main conclusions was that the methods of logical inference and search upon which AI relied were doomed to failure in anything but a controlled or limited environment. They pointed out that "in proving that one person could get into a conversation with another, we were obliged to add the hypothesis that if a person has a telephone he still has it after looking up a number in the telephone book." That is, in order to plan a sequence of actions, a computer must consider and account for every object in its universe since any one of them might ultimately be affected. For trivial agents like our hungry one, this is not a problem, but as an agent's universe is expanded, the task becomes absurdly difficult. A hungry agent with a scout at the door would starve to death if, in addition to a drawer full of money, it had a kitchen full of baking supplies, a store up the street, a bicycle with a flat tire, a telephone, a budget, a tight schedule, and a desire to give money only to charities representing its own interests. In the time a computer would take to plan its actions, the scout will have gone, the store will have closed, and any milk in the house will have gone sour. When you add the fact that anything but a trivial or highly limited problem area requires defining attributes with some degree of precision (the store closes at eight, is at the top of a very steep hill, has no bike rack, and is run by someone
who owes the agent money), it becomes obvious that the whole approach is seriously flawed.

McCarthy and Hayes referred to this as the "frame problem". They also pointed out how it was never really possible to foresee every consequence of every action. Exceptions are normal. A powerful computer might conclude that if the agent patches the tire and rides to the store it will find food. But what if the store had suddenly gone out of business, the road had been closed, or the patch failed? For every action, there are countless ways in which its consequences can be altered.

I could continue by citing numerous examples of AI applications throughout the 1960s and 70s, but this is not necessary. The most important point should now be obvious. Early enthusiasm for creating a "generally intelligent" computer quickly waned as people began to realize that even the fastest computers would have difficulty demonstrating it given even the most powerful AI techniques. Given this realization, AI researchers faced a number of choices. They could wait until computers became powerful and fast enough to handle such tasks, but this appeared to be measurable in decades or even centuries. They could essentially start from scratch and seek entirely new methods of processing, as some actually did by pursuing neural nets and parallel processing. Or, they could immediately apply the techniques that had been developed in the search for general intelligence to specific problems in relatively limited domains. It is this third option that we will now turn to for it turns out that far from being a cop out, it spawned a whole new area of research in what came to be called expert systems.

**Designing Medical Expert Systems**

In this section, we will look at some of the techniques and problems that are unique in designing expert systems. While most programs used in clinical medicine offer accounting, scheduling, patient monitoring, record keeping, or bibliographic retrieval capabilities, very few offer diagnostic assistance to practicing physicians [8]. The reasons for this stem primarily from the sheer complexity of the problem of actually building expert systems. The scope of medical knowledge and the complexities of inference in the clinical domain have made the development of medical expert systems a challenging
endeavor. Despite these challenges, the medical domain has proven to be one of the most fertile areas for expert systems research. Factors influencing current developments in medical expert systems include the development of domain specific inference algorithms, the availability of medical information, and the crisis that this information is creating within the health care field.

Formal techniques for reasoning with uncertainty are now available. Interest in such techniques began in the late 1960s when researchers began to realize the limitations of inference based on strict mathematical logic. Early enthusiasm for logical systems was tempered by the realization that few domains contain knowledge that can be explicitly encoded with mathematical certainty. All too often, inference is based on facts which have varying degrees of certainty attached to them. Nowhere is this more evident than in the domain of medical decisions where uncertainty is the rule. Techniques that were developed subsequent to formal logical systems include production rules incorporating probability theory and Bayesian methods. Many of these techniques were developed primarily within the field of medical expert systems and are thus quite specific in their utility. This domain specific development of inference algorithms has proven to be one of the biggest breakthroughs in AI and has ushered in the current era of applications which are the focus of this section.

The ready availability of medical knowledge in the form of published material in textbooks, medical journals, and other databases simplifies the task of knowledge engineering immensely. Medical knowledge engineers are more likely to face the challenge of having to decide what information not to include in a medical knowledge base than to have to go about creating new knowledge that bears on the problems of medical diagnosis. As we will see, the sheer size of the medical domain gives knowledge engineers an unprecedented amount of flexibility regarding how to conceptualize problems and formalize the representation of medical knowledge. Specifically, the granularity of information becomes an issue. That is, should inference proceed directly from signs and symptoms to disease states or by more subtle links between observations, pathophysiologic states and diseases. This is a design issue that is currently being investigated and which absolutely requires the existence of a well established body of medical knowledge. On the other hand, the fact that most medical knowledge continues to be disseminated in the form of paper-
based journal articles with little standardization of the knowledge base turns out to be one of the factors that also seems to limit the ability of developers to implement expert systems. In other words, there is plenty of knowledge, but most of it is in the wrong form. Medical knowledge tends to be in forms that must be abstracted, translated, transformed, or standardized to be useful. In fact, as we shall see, the requirements of medical expert systems are not necessarily identical to those of human experts meaning that in the future, we may need to augment our techniques of data collection and dissemination to facilitate the integration of automated techniques into the practice of medicine.

There is a downside to the size and rate of growth of the medical sphere of knowledge and that is that while these factors may well facilitate the development of medical expert systems, they are increasingly cited as the source of frustration for humans involved in medical decision making. The problems of the information age are unique. Whereas previous generations struggled to make decisions in the face of too little information, current practitioners must be extremely careful not to overlook information which is available but is out of reach either because of limitations in memory or in the technology of information retrieval. While text-based sources such as textbooks and medical journals still comprise the primary external source of information for medical professionals, they suffer serious drawbacks in the current environment of rapidly proliferating knowledge. They tend to be out of date, poorly organized, inflexible, inadequately indexed, and difficult and time consuming to attain and synthesize, especially as they relate to specific medical problems as they are encountered in everyday practice [9]. Thus, we find ourselves at a crucial turning point where traditional techniques for managing information are inadequate. Medical expert systems will gradually find their way into clinics and hospitals as the sheer amount of knowledge influencing medical decisions comes to exceed the capacity of the brightest most conscientious of physicians. As we shall see, the best medical expert systems are currently conceptualized as tools for clinical decision making, bringing problem specific knowledge within reach in seconds. It is this "knowledge at your finger tips" approach that fuels much of current enthusiasm for expert systems and promises to revolutionize the way we think about attaining problem-specific information in medicine.
Representing Knowledge in Expert Systems

Representing medical knowledge in a manner that can be understood by humans and machines alike is called knowledge engineering. It is separate from the problem of inference for several reasons. First and foremost, it is no small task. Medical knowledge is so vast and complicated that anyone who sets out to systematize it is making a major commitment in both time and money. Ideally, medical knowledge engineers are trained in computer science as well as medicine. Only then can they fully appreciate the tradeoffs and complexities involved in designing a knowledge base for the medical domain. To see how challenging this task is, let's imagine that we are creating a small knowledge base pertaining to just one medical symptom. Let's choose one symptom, abdominal pain localized to the right upper quadrant that is severe in quality, and trace the steps that are involved and the issues that are brought up during the process.

First, we need to name our symptom in a way that is understandable to those who will interact with the system, namely physicians and nurses who will use the system, and the knowledge engineers and computer scientists who will be responsible for developing the system and creating the inference engine. It makes sense to call the symptom Severe Right Upper Quadrant Pain, since most medical professionals have some sense of what this means. We might like to condense the term and express it as "Severe RUQ Pain" to save space and make for more rapid scanning and interpretation.

But immediately we are faced with a question. Is Severe RUQ Pain a single fact or is it several facts in a relation. Are we talking about a discreet event (pain in the area just below the ribs on the right side that feels severe)? Or do we wish to depict this as a feeling of pain (one of several permitted feelings including nausea, tingling, or heat) located in the right upper quadrant (which we abbreviate RUQ and which is just one of many locations on the human body) having a quality that is severe (one of several varieties of pain which include dull, throbhing, aching, excruciating). As trivial as they may seem, such decisions are extremely important and fundamental in the domain of knowledge engineering. They deal with the granularity of the information that we seek to represent.
There is currently no single standard for representing medical knowledge. Rather, each system incorporates one or more elements of several methods which are depicted here. A few authors have suggested that there should be a standard medical knowledge syntax and semantics and some have even proposed their own prototypes. The advantages of using such a standard include the ability of various systems to share a common knowledge base. If a single standard is ever adopted, it will be possible to maintain immense knowledge bases containing the contributions of numerous researchers. However, it is probably too early to adopt any particular standard as we are not yet certain as to the optimal strategy for medical knowledge representation. More experiments are needed so that we can compare various strategies to decide which are best.

**Propositional Logic**

Propositional logic is representation language in which statements are either true or false. Statements such as the following can be represented in propositional logic:

- The patient has a fever
- The patient's liver is enlarged
- The patient has severe RUQ pain

These statements are propositions. Each can be either true or false. Given their compactness, we can call such statements *atomic propositions* and even abbreviate them with letters F and L:

- \( F = \) The patient has a fever
- \( L = \) The patient's liver is enlarged

The truth or falsehood of each proposition is defined by an interpretation function, \( \omega \), which assigns each proposition in a set of propositions a value of either true or false. Thus:

\[ \omega(P) = \text{true} \]

asserts that the statement "the patient has a fever" is true.
Atomic statements such as these can be combined using logical connectives to form composite propositions. In propositional logic there are five connectives. These are outlined in Table 3.

<table>
<thead>
<tr>
<th>Connective</th>
<th>Meaning</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Negation</td>
<td>Not</td>
<td>(\neg)</td>
</tr>
<tr>
<td>Conjunction</td>
<td>And</td>
<td>(\land)</td>
</tr>
<tr>
<td>Disjunction</td>
<td>Or</td>
<td>(\lor)</td>
</tr>
<tr>
<td>Implication</td>
<td>If then</td>
<td>(\rightarrow)</td>
</tr>
<tr>
<td>Bi-implication</td>
<td>If and only if</td>
<td>(\leftrightarrow)</td>
</tr>
</tbody>
</table>

Thus, the composite proposition:

\[ P \land Q \]

has the meaning:

The patient has a fever and an enlarged liver.

By combining logical connectives, more complicated statements can be generated. A set of logical rules defines the relationships between propositions with known interpretations and connectives consisting of them. Table 4 shows these relations.

<table>
<thead>
<tr>
<th>(\omega(P))</th>
<th>(\omega(Q))</th>
<th>(\omega(\neg P))</th>
<th>(\omega(P \land Q))</th>
<th>(\omega(P \lor Q))</th>
<th>(\omega(P \rightarrow Q))</th>
<th>(\omega(P \leftrightarrow Q))</th>
</tr>
</thead>
<tbody>
<tr>
<td>true</td>
<td>true</td>
<td>false</td>
<td>true</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>false</td>
<td>false</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>false</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>true</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>false</td>
<td>false</td>
<td>true</td>
<td>false</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
</tbody>
</table>

Using propositional logic to designate associations in the medical domain would go as follows:
Suppose you wanted to express the conditions necessary for diagnosing meningitis. Typically, clinicians suspect meningitis in patients having fever, headache, lethargy, confusion, irritability, and stiff neck. Each of these symptoms can be treated as a proposition as follows:

\[
\begin{align*}
F &= \text{the patient has fever} \\
H &= \text{the patient has a headache} \\
S &= \text{the patient has a stiff neck} \\
L &= \text{the patient is lethargic} \\
C &= \text{the patient is confused} \\
I &= \text{the patient is irritable}
\end{align*}
\]

Now, when we consider that \(L, C, \) and \(I\) are relatively soft signs, we might loosen our criterion and diagnose meningitis in patients having the first three signs and only one of the last three. Thus, the diagnostic algorithm can be expressed succinctly as:

\[
(F \cap H \cap S) \cap (L \cup C \cup I) \rightarrow \text{Meningitis}
\]

That is, if the patient has a fever, a headache, and a stiff neck and is either lethargic, confused, or irritable, then the patient has meningitis.

Note that we have hedged somewhat. We might have chosen to write our algorithm as follows:

\[
(F \cap H \cap S) \cap (L \cup C \cup I) \rightarrow \text{Meningitis}
\]

indicating that only meningitis causes these signs and symptoms and that meningitis only occurs when they are present in the stated relation. Clearly, this would be a mistake. Some patients with meningitis don't have fever, for example, while others with fever, headache, a stiff neck, and lethargy don't have meningitis. In fact, medicine is typically characterized by such uncertainty. Few relations are certain. Signs and symptoms which are diagnostic of a particular disease are called pathognomonic and are characterized by a bi-implicational connectives as in the above statement which is clearly false. In actuality, pathognomonic signs and symptoms are quite rare in medicine. Rather, the presence or absence of a finding is usually associated with a degree of implication of one or more diseases.

It is not too hard to imagine ways around this using propositional logic. For example, one might express the conditions implying meningitis in a number of ways and "or" them together. This is reasonable, but impractical for most purposes. Meningitis, like most diseases, can present in numerous ways and it
is virtually impossible to account for all of these ways in an algorithm that is purely logical. To do so would require anticipating and encoding every possible presentation of meningitis explicitly. Moreover, even if we could account for all possible presentations of meningitis, logic gives us no way of ranking our hypotheses. That is, by ruling meningitis in or out, we leave no room for ambiguity. It would be nice if there was some method for assigning numerical values to our hypotheses based on our certainty in them.

*First-Order Predicate Calculus*

As you may have noted, propositional logic is somewhat limited in that it is incapable of expressing abstractions and generalizations. Facts must be explicitly stated as in: "the patient has RUQ pain." A more expressive representation language is *first-order predicate logic* which allows for the construction of logical statements about objects prior to their explicit denotation. For example:

\[ \forall x \ (\text{Meningitis}(x) \rightarrow \exists y \ (\text{Virus}(y) \cup \text{Bacteria}(y)) \cap \text{Etiology}(y,x)) \]

states that all meningitis is caused by either a virus or a bacteria \(^1\).

*Medical Reasoning is Probabilistic*

Diagnosis (di"ag-no"sis) [Gr. dia through + gnosis knowledge]. When a physician looks at a sick patient, he attempts to see everything that is relevant in order to come up with a list of diseases that could possibly cause his particular constellation of signs and symptoms. As in the above example, the signs fever, headache, stiff neck, confusion, irritability, and lethargy strongly suggest a diagnosis of meningitis. But meningitis does not always present in the same way. Sometimes, a patient will present with a subset of these signs and symptoms. Other times, a patient might exhibit atypical signs. Moreover, there are more specific observations such as analysis of cerebrospinal fluid that will rule in or out the diagnosis of meningitis with a high degree of certainty but which are more expensive and invasive than asking questions or taking a

\(^1\)For all \(x\) that are meningitis there exists a \(y\) that is either a virus or a bacterium and is the etiology of \(x\).
temperature. Putting together the various signs and symptoms and knowing what to look for next is generally regarded as part of the art of medicine. Physicians are trained in anatomy, physiology, biochemistry, and pathology so that they can reason from signs and symptoms to produce a list of possible diagnoses and then choose more sensitive tests to narrow in on the actual diagnosis. This is the diacritical method that every medical student learns and which is the basis of most diagnostic reasoning in medicine to this day.

A patient with a fever, who is confused and has a headache may have meningitis. By asking the patient to touch his chin to his chest, the physician may or may not observe a stiff neck. If the patient's neck is indeed stiff, the physician may immediately proceed to a more specific test. But what if the patient's neck is not stiff? How does this affect the physician's assessment of the probability of the patient having meningitis? It is intuitively obvious that it diminishes it. But by how much? How does the physician rank the probability of the various diseases in his differential?

The answer is that most physicians proceed in a somewhat simplified version of an algorithm which we will formalize momentarily [10-12]. The physician should have a rough idea of what the most common causes of the signs and symptoms are and should know how to rule them out. He should also know some of the rarer but more dangerous causes as well. Attacking the most common causes first and proceeding to the rarer ones is a reasonable strategy and it usually works.

But there are some problems with this strategy. First of all, it relies on physician's memory. Given that some very deadly but treatable diseases are indeed quite rare, it is not uncommon for physicians to overlook them at considerable cost in terms of morbidity and mortality. When time is of the essence, it is therefore often not reasonable to assume that rare diseases can be considered after all of the more common ones have been ruled out. Another consideration is the cost of "working up" each disease. If one disease can be ruled out with a non invasive and relatively inexpensive test, it may take precedence over another disease that is diagnosed via a more expensive or dangerous modality.
Balancing these various factors is often not difficult as when a common disease presents commonly. But in other instances, it is extremely complicated. Sometimes patients die of treatable diseases that were never considered by physicians. More often, physicians jump prematurely to expensive or invasive tests before performing maneuvers that are cheap and easy. In the present environment of cost containment and quality control, it is no longer enough to know which tests to order. Now more than ever, the physician is expected to know what order in which to test. Given that such a decision is multifactorial and numerically complex, it stands to reason that in the future, competition on the basis of cost and quality will demand a more accurate and efficient method that is some cases may actually require machines. In the next few sections, we will review some of the analytical methods that are available for performing such analyses.

Sensitivity and Specificity

Two measures that help physicians determine the value of information used to make a diagnosis are sensitivity and specificity. These measures are treated simultaneously because in many ways they are complimentary.

Sensitivity is a measure of how well a test or a finding can "pick up" a particular disease. Its definition is:

\[ \Pr(\text{symptom} \mid \text{disease}) \]

That is, sensitivity is the probability that the symptom will be present given a particular disease. Specificity, on the other hand, is the probability that the symptom is not present given that the person does not have the disease:

\[ \Pr(\text{no symptom} \mid \text{no disease}) \]

Specificity is a measure of how confident you can be that a disease is not present given the absence of a particular finding. To see how these numbers might be used, let us return to our example of meningitis.

Suppose we sampled one thousand patients and assessed each one for the presence or absence of meningitis and a stiff neck. We could then derive a table such as table 5.
Table 5. Frequency Table Showing Patients With and Without Meningitis and Stiff Neck

<table>
<thead>
<tr>
<th>Meningitis</th>
<th>No Meningitis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stiff Neck</td>
<td>36</td>
</tr>
<tr>
<td>No Stiff Neck</td>
<td>4</td>
</tr>
</tbody>
</table>

132 868
40 960 1000

Using the data in table 5 we can derive the sensitivity and specificity of a stiff neck regarding meningitis:

Sensitivity = Pr (finding | disease) = 36/40 = 0.90

Specificity = Pr (no finding | no disease) = 48/960 = 0.05

Which is to say that a patient with meningitis has a 90 percent chance of having a stiff neck but that five percent of patients without meningitis are likely to have the same symptom.

Getting a computer to calculate statistics like this is trivial. However, as we have already discussed, there is a lot more to medical diagnosis than assessing the probability of disease given one symptom. How might this sort of analysis be extended to account for a number of symptoms?

Bayes' Theorem and Posttest Probability

The answer is to use Bayes' theorem which updates the probability of having a particular disease given its pretest probability. After assessing a number of signs and symptoms, it is possible to assign probabilities to one's diagnostic hypotheses. The addition of a new finding adds information by either raising or diminishing the likelihood that each hypothesis is true. The derivation of Bayes' theorem is relatively simple and can be found in most introductory textbooks on statistics. Its result is quite useful and states that the probability of a disease given a particular finding:

$$\Pr(D|F) = \frac{\Pr(D) \times \Pr(F|D)}{\Pr(D) \times \Pr(F|D) + \Pr(-D) \times \Pr(F|-D)}$$

Where:
\[ Pr(D) = \text{Pretest probability of disease} \]
\[ Pr(F|D) = \text{Probability of finding given disease} \]
\[ Pr(-D) = 1 - Pr(D) \]
\[ Pr(F|-D) = \text{Probability of finding given no disease.} \]

A number of assumptions are made when using Bayes' theorem. One is that all of the findings one is accounting for are *conditionally independent*. This means that the probability of one finding is not affected by the presence or absence of any other. In reality, this assumption is often violated. For example, when a patient has a stiff neck, the probability that he also has a fever is increased since both findings are related to meningitis. It turns out that there are some ways of getting around this, but we will cover these when we get to the specifics of some of the expert systems later in this chapter. The main point is that Bayes' theorem is a statistical technique for calculating the probability of a particular disease given a new finding. Given sufficient data (in the form of tables such as table 5 for each sign as it relates to each disease) one could conceivably assemble a computer program that iteratively calculates the probability of a number of diseases as the presence or absence of each finding is entered. As it turns out, the systems we will review rely on such statistics despite the fact that they are rarely explicitly calculated. This means that the developers of expert systems must rely on *estimates* which may be one limitation of this method.

**Constructing a Knowledge Base**

Given these tools for depicting medical knowledge, knowledge engineers can encode medical knowledge in various formats. A number of explicit strategies have been derived.

**Rule-Based Systems**

One method is to construct a set of interrelated rules that can be used to guide a system from data to conclusions. For example, figure 1 depicts an algorithm that might be part of a system used to diagnose meningitis.
Rule-based methods such as this typify early medical expert systems. Knowledge was explicitly encoded in a series of rules that guided inference. While systems based on these methods did prove successful in some cases, rule-based systems have a number of drawbacks. First, they do not facilitate the easy application of statistical techniques as described earlier. Second, and more importantly, they couple inference with knowledge in a way that makes the addition of new knowledge a cumbersome and complicated problem. The addition of new knowledge changes the algorithm since that is where the knowledge is encoded. As the algorithm grows in complexity, the addition of a single fact can become quite intimidating.
Frame-Based Knowledge

One way of getting around these problems is to depict medical knowledge in frames. A knowledge frame is a set of information that is related in some way and which can be "plugged in" to a system without too much effort. Table 6 depicts a knowledge frame for meningitis as discussed previously.

Table 6: A Hypothetical Knowledge Frame for Meningitis

<table>
<thead>
<tr>
<th>Finding</th>
<th>Sensitivity</th>
<th>Specificity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fever</td>
<td>0.90</td>
<td>0.15</td>
</tr>
<tr>
<td>Stiff Neck</td>
<td>0.00</td>
<td>0.06</td>
</tr>
<tr>
<td>Headache</td>
<td>0.85</td>
<td>0.35</td>
</tr>
<tr>
<td>Confusion</td>
<td>0.60</td>
<td>0.20</td>
</tr>
<tr>
<td>Lethargy</td>
<td>0.55</td>
<td>0.25</td>
</tr>
<tr>
<td>Irritability</td>
<td>0.50</td>
<td>0.40</td>
</tr>
</tbody>
</table>

Note that we have arbitrarily chosen to depict the sensitivity and specificity of each of our findings for meningitis. This suggests that we may perform some kind of calculation later on using this data. In fact, we could just as well have included other relevant statistics depending on our needs. For example, we might include various organisms that cause meningitis and their frequencies or some syndromes that predispose patients to meningitis and the degrees to which they do so. There are no rules about what data is relevant in a disease frame. It depends on how you plan to use the knowledge once it is encoded. For example, sometimes it is not feasible to use actual statistics and one is forced to make estimates. Another important consideration is whether to use raw data or statistics that have been derived from it. The tradeoff between these options is primarily between speed and flexibility. Raw data or lower level statistics allow derivation of new statistics down the road but this takes time and slows down processing. Having ready access to statistics in memory speeds up computing by reducing the amount of processing that the machine must do each time it performs its function.

Medical expert systems began to receive attention beginning in the early 1970s. As you will recall, this is around the time the development of generally intelligent systems began to appear more ambitious than had previously been assumed. Despite these failures, AI had indeed developed an arsenal of techniques for manipulating symbolic data by this time. Therefore, the transition to smaller, more well circumscribed domains was a natural one.

Of particular historic interest are developments at four major institutions between 1970 and 1990, Massachusetts Institute of technology, Pittsburgh University, Rutgers, and Stanford.

In 1974, Edward Shortliffe was a doctoral student at Stanford University. He was interested in creating a rule-based system that would aid physicians in deciding how to treat infectious diseases. The emphasis of MYCIN [13], the system he created to do this, was therefore on advising rather than diagnosis. The distinction is minor, however, as the idea was to encode expert knowledge about infectious disease so that it could be accessed rapidly when needed.

The problem that MYCIN was designed to solve was as follows: The diagnosis of infectious disease is complicated. The signs of various diseases are often quite similar, and distinguishing between them usually requires observation of the etiologic organism in a laboratory culture. Unfortunately, culturing an organism takes time. There is typically an interval during which therapy must be instituted in the absence of a definitive diagnosis. The optimal strategy is therefore to base the choice of antibiotics on specific signs and risk factors that point to one or more microorganisms and then wait for more definitive data from the lab. When cultures come back, antibiotic therapy can be adjusted to be more specific. Relying on one's memory to make such preliminary determinations is admirable but in general this strategy is prone error. It is simply too easy to forget one fact and endanger patients in this manner. A better strategy is to incorporate all of the relevant knowledge into an algorithm and access that algorithm when necessary. This way, the physician can be sure that he is not leaving anything out of his differential and thereby omitting an essential therapy.
MYCIN was a rule-based system that represented knowledge in a form such as that depicted in figure 2.

**Figure 2. A Typical Rule as Seen in the MYCIN System (Source: Buchanan, et. al. [13])**

<table>
<thead>
<tr>
<th>RULE 156</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF:</td>
</tr>
<tr>
<td>1. The site of the culture is blood, AND</td>
</tr>
<tr>
<td>2. the organism stains gram-negative, AND</td>
</tr>
<tr>
<td>3. the organism is rod shaped, AND</td>
</tr>
<tr>
<td>4. the organism gained entry via the urinary tract, AND</td>
</tr>
<tr>
<td>5. the patient has not undergone any genitourinary procedures, AND</td>
</tr>
<tr>
<td>6. the patient has not been treated for cystitis</td>
</tr>
<tr>
<td>THEN:</td>
</tr>
<tr>
<td>There is evidence that the organism is e. Coli.</td>
</tr>
</tbody>
</table>

MYCIN used *production rules* to reason from observed data to conclusions. This form of reasoning is known as *data driven* or *forward chaining*. The inference algorithm utilizes those rules containing information that is known and then chains forward to inquire about certain facts which would facilitate further inference. For example, rule 156 (figure 2) might be flagged if a blood culture revealed gram-negative rods. MYCIN might then inquire about the remaining facts by invoking an interactive session with the user in which he was asked to supply the missing information, i.e. whether the patient had a urinary tract infection, had undergone any procedures, or had been treated for cystitis.

As you can see, the distinction between is not entirely cut and dried. MYCIN’s knowledge base incorporates elements of both rule-based and frame-based knowledge thereby retains some the advantages of each, including the ability to easily change the knowledge base without having to alter the entire inference algorithm. The production rules can be left intact while the knowledge frames are manipulated at will. Another advantage of the MYCIN approach was that the reasoning strategy was explicit. By retaining an English language version of each rule, MYCIN was made capable of displaying its logic. This turns out to be an important factor as physicians tend to be skeptical of devices that give advice without explanation.
MYCIN was evaluated in two studies and was found to give advice that was deemed comparable to that one would expect from an infectious disease expert. However, limitations in the power of computers in the 1970s prohibited the cost-effective use of MYCIN in an actual clinical setting. Despite its limitations, Shortliffe's experimental program captured the attention of AI researchers within and outside of the medical field. During the next two decades, a number of research teams worked on medical expert systems at various institutions. Figure 3 depicts the time course of a number of these projects.

**Figure 3. University Medical Expert Systems Projects 1970-1990**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MIT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pittsburgh</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rutgers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stanford</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Rather than covering each of these systems superficially, it is useful to pick out two that have achieved a measure of commercial success. A review of the literature reveals that only a few expert systems have managed to "break through" the academic barrier to become accepted by consumers of medical software. Among these are QMR and Illiad. In the next two sections, we will look at these systems in order to get an idea of how they work and how well they perform. Then, we will compare them and move on to some of the implications of expert systems for the future.
Quick Medical Reference and Iliad: State of the Art Systems

Internist & Quick Medical Reference

Internist-I [8, 14-22] is a computer based diagnostic consultant for general internal medicine. Quick Medical Reference (QMR) is the name given to a newer, revised version of Internist. The program uses some of the symbolic reasoning strategies of artificial intelligence to explain observed findings in terms of differential diagnoses. The basic idea is to enter a set of findings (symptoms, signs, laboratory values) and to generate a list of diseases that could explain each finding and then to process the list intelligently to arrive at the most likely diagnosis or diagnoses.

Development of Internist-I and QMR began in 1972 by Myers, Pople and Miller at the University of Pittsburgh School of Medicine. The original idea was to develop a diagnostic computer algorithm that could mimic the reasoning of an expert clinician.

The Internist Knowledge Base

Internist-I incorporates a frame-based approach. Associated with each disease in Internist's knowledge-base is a list of associated findings. Inverting this list produces a list of diseases associated with each finding (the differential diagnosis of each finding). As findings are entered into the system, Internist generates a comprehensive list of diseases that could explain them (forward chaining). If this were the only step, Internist-I would have done little more than generate long lists of diseases, each of which might explain one or more of the findings in a particular patient. The intelligent aspects of Internist-I's performance depend on the processing of several other variables associated with each finding in a disease frame. Thus, under a disease is a list of findings and associated with each finding is a list of variables that quantify the association of that finding with the disease:

The variables associated with each finding are Evoking Strength, Frequency, and Import. Evoking Strength is a measure of how strongly the disease is suggested by the finding and is closely related to specificity. Evoking Strength is not derived through scientific investigation since exact calculation of specificities for all of the manifestations of all of the diseases in Internist-I's data.
base is unrealistic. Rather, it is assigned a value on an ordinal scale ranging from zero (the finding is too nonspecific to point to the diagnosis) to five (the finding is pathognomonic for the disease) based on review of the medical literature and the judgments of expert clinicians. *Frequency* is a measure of how often the finding is associated with the disease. It too is derived through clinical judgment and it is assigned a value between one (the finding is a rare manifestation of the disease) and five (the finding occurs in all cases of the disease). Finally, each finding is assigned an *Importance*. This is a measure of how important it is to explain the finding. Import ranges from one (the finding is unimportant and may be ignored, e.g. malaise) to five (the finding must be explained by the final diagnosis, e.g. jaundice).

The Internist-I knowledge base has grown over the years to include over 572 disease profiles, about 4000 manifestations of disease, and more than 4,000 "links" detailing the causal, temporal and probable interrelationships among the disorders. The disease profiles were derived under the supervision of JD Myers, M.D.. Diseases were divided among contributors who conducted extensive literature surveys to create lists of findings that have been reliably reported to occur in patients with each illness. Such findings included demographic, epidemiologic, historical, physical, and laboratory data. A typical disease profile contains an average of 85 findings (range, 25 to 250 findings). In addition to findings, "linked diagnoses" are also identified. Linked diagnoses are those conditions predisposing to the disease, causally related to the disease, occurring more often than chance in association with the disease, or temporally preceding the development of the disease.

In addition to the disease profiles, Internist-I includes the profiles of several "high-level pathophysiologic states" such as chronic uremia, left-ventricular failure, and renal failure. By constructing links between these intermediate states and their underlying causes, the designers of Internist-I hoped to approach diagnosis by working from findings through states to disease. The states would serve as logical bridges between findings and diseases. By bundling findings into known syndromes with well established pathophysiology they hoped to facilitate the use of known pathophysiologic mechanisms stored in its database.
The Internist-I Partitioning Algorithm

The original Internist-I algorithm was written in LISP and assembly language for a large mainframe. The algorithm works like this:

The user enters all of a patients positive and negative findings. Internist-I then generates a list of all possible diagnoses that could explain any of the patient's positive findings. Each of these "disease hypotheses" has attached to it (1) a list of all findings explained by the disease, (2) a list of all findings not explained by the disease (3) a list of all of the findings that would be expected with the disease but weren't present in the patient, (4) a list of all findings of the disease about which nothing is known.

The disease hypotheses are then individually scored. The scores are broken into positive and negative components:

The positive component is a nonlinear weighted sum of the evoking strengths of the patient's findings with a bonus for any links to previously established diagnoses. The negative component is a nonlinear weighted sum of the frequencies of each finding expected in the disease but not found, in this patient and of the findings present in the patient but not found in the disease. The overall score is the sum of the positive and negative components. Disease hypotheses are then sorted by score and those with scores below a threshold are set aside.

A partitioning algorithm, developed by Harry Pople, separates the hypothesized diagnoses into problem areas. Two diseases are competitors (and belong in the same problem area) if together they explain no more of the patient's findings than one or the other diagnoses taken alone. Diagnoses are not competitors (and therefore belong in different problem areas) when they explain more findings together than when taken alone. The rest of the algorithm attempts to resolve each problem area to one diagnosis.

If the "best" hypothesis for a particular problem area is 90 or more points "better" than the second best, then it is chosen as the sole diagnosis. If not then Internist-I selects a questioning strategy. When hypotheses 1 and 2 are 46-89 points apart, Internist-I adopts a pursue strategy which is to ask questions that are most specific for the leading disease hypothesis. When five or more
diagnoses lie within 45 points of the leading diagnosis, Internist-I goes into ruling out mode and asks questions about findings that are expected to occur with high frequency for the disease hypotheses hoping that a few negative answers will remove some of the diagnoses from contention. A discrimination mode is used when there are two to four diagnoses within 45 points of the leading diagnosis. Internist-I asks questions in groups according to the traditional order of collecting clinical data beginning with the history and then progressing on to the physical exam, labs, and confirmatory tests.

When a diagnosis is achieved, Internist-I recycles using those findings not yet explained by the diagnosis. The program continues until there are no more useful lines of questioning or until all remaining manifestations have import of two or less.

Quick Medical Reference

In 1985, problems with the Internist-I approach [23] to diagnosis led to the development of QMR. The role of Internist-I had been solely to generate differential diagnoses. QMR retained the knowledge base developed for Internist-I and used the partitioning algorithm described above. The addition of a number of features expanded the flexibility of the system making it more like an "electronic textbook" of medicine [17]. QMR was written in Turbo Pascal to run on IBM personal computers.

Key features of the QMR program include:

The QMR completer which solves the problem of users having to adhere to a strict medical taxonomy of findings. The user has only to type the first few characters of a finding or even just an abbreviation for the finding. The completer then displays a menu of all of the findings that could be signified by the abbreviation allowing the user to select those which apply.

A low-level information retrieval tool allows the user to access the information stored in the QMR knowledge base. The user can ask QMR to display the findings and links associated with a disease or conversely to display the differential diagnosis of a specific finding. An intermediate-level information management tool allows the user to limit the differential diagnosis to diseases of specific organ systems.
Case analysis mode allows the display of those diagnoses consistent with a patient's findings as in the Internist-I approach with a few additional features including the ability to identify closely related diagnoses which in combination could explain a patient's findings. In this manner, QMR can offer various "global overviews" which are logical explanations of the patient's findings given one or more interrelated diagnoses. The user can even run the case analysis mode after pulling out a particular diagnosis known to exist in the patient in order to allow QMR to explain just those findings not related to an established diagnosis.

Critiquing mode allows the user to enter his diagnostic hypothesis and have QMR display lists of those findings which support it, those that are not supported by it and those which are specifically contradicted by available evidence.

Together, these features extend the capability of the Internist-I/QMR system to that of an interactive diagnosis and information retrieval system.

Evaluating Internist-I and QMR

Internist-I was developed using CPC's from Case Records of the Massachusetts General Hospital from the New England Journal of Medicine. One of the earliest evaluations of Internist-I [15] involved the use of NEJM CPCs that had not been used in the design phase. Internist-I was presented cases from the NEJM that had in their differentials only diseases that were in Internist-I's incomplete knowledge base. The results of this evaluation showed that Internist-I failed to make the correct diagnosis 42% of the time compared to clinicians who failed 35% of the time and discussants who failed in 19% of cases. Internist-I made fewer incorrect diagnoses, however, suggesting incorrect diagnoses in just 26% of cases compared with 30% for clinicians and discussants alike.

The authors discussed the major limitations of Internist-I and included the inability of Internist-I to reason anatomically or temporally. Findings were evaluated solely on the basis of their specificity, frequency, and importance. Other factors might include their anatomic, temporal and causal relations to other known positive findings. The authors suggest refining Internist-I by breaking its knowledge base down on the basis of pathophysiologic states that can be linked on the basis of known causation or association.
One of the concerns early on in the development of QMR and Internist-I was that variations in the ways people input data might affect performance. Those who were using QMR in 1987 were typically experts who had been involved in the development of the system and knew its capabilities quite well. It remained to be seen whether non-experts could use the system successfully with minimal training. One study [19] compared the use of QMR by experts and novices and found that there was significant variation between the findings that they chose to enter. Novices entered only 60% of the positive and 30% of the negative findings that were entered by experts suggesting that some training might be required to make proficient users of the system. Some of the problems suggested by the authors of this study were QMR's inability to process temporal information, variation in choosing precise vocabulary terms, and differences in interpretation and entering of graphic data. It is noteworthy that the average time to enter a case was about one hour and that such a time commitment is unreasonable.

In 1989, Bankowitz and colleagues [8, 20] studied the effect of QMR on the diagnostic strategies of housestaff. They employed a pre- and post-consultation questionnaire method to determine whether QMR aided in the diagnosis of challenging cases in internal medicine and found that overall, 58% of consultations resulted in changes in the differential diagnosis consistent with the recommendations of QMR. Changes included the addition (45%), elimination (26%) and reordering (23%) of diagnoses. Consultations with QMR also had an impact on diagnostic management resulting in the ordering of new diagnostic studies, the elimination of studies, or changes in the order of studies in 42% of cases. Users were also asked to rank the usefulness of the system. Using a categorical scale, 81% rated the system as being useful, 16% were neutral, and 0% said that it was not helpful (3% did not answer this question).

The Iliad Clinical Consulting System

Iliad [24-34] was designed for the Macintosh operating system and has a completely different feel from QMR. Iliad uses pull down menus, dialogue boxes and windows, the familiar Macintosh interface, to guide the user through sessions involving the diagnosis of illness or simulation of a patient for educational purposes. The Iliad expert system was designed to support medical problem solving in the field of internal medicine at the University of
Utah at Salt Lake City and is available through Applied Informatics (295 Chipeta Way, Salt Lake City, UT 84108; 801-584-3060.) Iliad is an offshoot of the HELP (Health Evaluation Through Logical Processes) hospital information system reviewed in the next chapter.

An Overview of Iliad

Iliad's knowledge base covers 908 diseases, their treatment protocols, references to the medical literature, ICD-9 codes, and the costs for medical tests. Each disease is associated with a structured database of 11,900 disease manifestations covering internal medicine and those aspects of pediatrics, dermatology, psychiatry, OB/GYN and sleep disorders relevant to the typical internist. These links are in the form of frame-based Bayesian probabilities which are used to drive inference during diagnostic sessions. Currently Iliad is one of the best known of all of the commercially available medical expert systems. Like QMR, Iliad has been demonstrated at numerous conferences and is currently available commercially in Version 4.2 at a cost of $995. It is in use at over one-half of the medical schools in the country and by over 800 practicing physicians [33].

Iliad has several modes. In consultation mode, the user enters observations and Iliad displays a differential diagnosis. In critiquing mode the user enters his presumptive diagnosis and Iliad displays a list of findings not explained by that diagnosis as well as the specific information required to establish it. In browse mode the user can search through the knowledge base and access treatment protocols, bibliographic citations, and abstracts from the Year Book of Medicine. Finally, there is a simulation mode in which the user works through a simulated case by requesting information much as he would do in a real clinical situation. Iliad scores the user's performance, calculates the cost of the workup, and gives specific feedback on request. Future versions of Iliad will support access to video laser-discs, thereby providing actual views of x-rays, echocardiograms, skin lesions, and other physical findings aiding both consultation and simulation modes tremendously [35].
Iliad’s Knowledge Base

Iliad's knowledge base was developed by a panel of specialists in cardiology, pulmonology, and gastroenterology with additional training in medical informatics. These experts have held over 8,000 one or two hour sessions since the Iliad's inception to bring the knowledge base to its current status of containing 2,500 disease frames, 908 diagnoses, 1,509 syndromes, and 11,910 disease manifestations [33, 36]. The basic unit of knowledge in Iliad is the disease frame. A disease frame connects a disease or a pathophysiologic state to a list of associated clinical findings. There are two kinds of disease frames in Iliad's knowledge base, Bayesian and Boolean.

Boolean frames contain a number of findings that are conditionally dependent and stem from the existence of a single pathophysiologic state. Clusters of findings are associated with a syndrome through Boolean logic. An example would be a cluster called Hypoxemia which is defined by the presence of either an arterial PO2 of less than 60 mm Hg or the existence of cyanosis [33]. Thus, establishing the presence of a pathophysiologic state requires satisfying a logically specified subset of requirements. There are two advantages of having Boolean disease frames describing pathophysiologic states. First, it is a way of getting around the fact that Bayesian inference assumes that all findings are mutually independent. Second, it facilitates a more rational approach to inference and the explanation of diseases in terms of their underlying mechanisms.

Bayesian frames relate diseases to associated findings and pathophysiologic states via a number of statistics: (1) The prevalence of the disease; (2) The sensitivity of each finding (probability of the finding given the disease); (3) An estimate of one minus the specificity (the probability of the finding given the absence of disease); (4) The likelihood ratio (the sensitivity divided by 1-specificity) which is used to measure information gain; (5) The cost in dollars of each finding.

Iliad’s Inference Engine

As you probably guessed from the structure of Iliad's knowledge base, inference proceeds by a combination of Boolean and Bayesian logic. Iliad
marks the presence or absence of all of the findings entered by the user and then processes the statistical information to derive the probability of each disease in its knowledge base through a forward-chaining algorithm.

Iliad uses utility function to determine which of the findings that are not known would contribute the most information for the lowest cost. The idea is to give Iliad the means of suggesting the most cost-effective strategy at any point in a workup. In 1992, Iliad used the following utility function [31]:

$$\text{Utility}(F) = \frac{\text{Prob} \times \text{Gain}}{\text{Cost}}$$

$$\text{Gain} = \max \{X \times (\text{sen}/(1-\text{spec})), Y \times ((1-\text{sen})/\text{spec})\}$$

Where:
- Prob = the prior probability of the frame being true
- Sen = the true positive rate, P(F|D)
- Spec = the true negative rate, P(F|nD)
- X = the degree of closeness to being true of a finding or a cluster in a disease frame.
- Y = the degree of closeness to being false of a finding or a cluster in a disease frame.
- Cost = the dollar cost to acquire the finding. The cost used in most cases is the actual charge at the University of Utah medical center. History and physical exam items are set to a cost of $1 and $2 respectively.

The algorithm runs through each diagnostic hypothesis and calculates the utility of each associated finding. For example, given a differential with three diseases and a total of ten unknown findings, Iliad would calculate 30 utilities. The finding with the highest utility is available by selecting the "next best information" function. It is important to point out that this function calculates the utility of each finding only within the context of a single disease hypothesis. Thus, situations where a finding might simultaneously contribute information about multiple disease hypotheses are not considered [31].

In 1992, Di Guo and colleagues [31] compared this utility function to four alternatives. Two of the alternative models came from information theory and two were what the authors called "quasi-utilities" which they derived on their own.
One of the utility functions was called the "logP2-logP1" model. This model was based on a measure proposed by Johnson [37] wherein uncertainty (H) is represented by the negative log to the base-two of the probability (P) of an event:

\[
(3) \quad H = -\log_2 P
\]

In this model, uncertainty is best conceptualized as the quantity of information required to achieve certainty given the current probability of an event. The quantity of information gained on learning the outcome of a test or a finding (F) is given by:

\[
\Delta H = \log_2 P(D|F) - \log_2 P(D).
\]

If

\[
I(D|F_K) = \text{abs}(\Delta H)
\]

given a known finding (F_K), then the absolute information gain can be calculated for both positive and negative findings:

\[
I(D|F+) \text{ and } I(D|F-).
\]

The model chooses the greater of these two values which is expressed:

\[
I(D|F) = \max\{I(D|F+), I(D|F-)\}.
\]

A second model from information theory is based on the concept of "entropy" defined by Shannon [38]. This model is similar to the previous model and represents the uncertainty associated with a particular set of diseases (D_1 through D_n) by:

\[
H(D) = - \sum P(D_i) \log_2 P(D_i).
\]

Guo implemented a limited version of this model wherein only one disease was considered at a time and the information gained by the presence or absence of a finding was given by:

\[
I(D|F_K) = \text{abs}(H(D) - H(D|F_K)).
\]
One problem with this conception of information gain occurs when the prior and posterior probabilities are complementary (i.e. the prior is 0.1 and the posterior is 0.9). In such a situation, the information gain in equation (7) would be zero which is counterintuitive. To get around this, the authors came up with an alternate expression to be used instead of (7) whenever the transition between prior and posterior probability passes through a maximum:

$$I(D|FK) = (H_{max} - H(D)) + (H_{max} - H(D|FK))$$

where:

$$H_{max} = 1.$$ 

The first "quasi-utility" measure of information gain tested by Guo was called the "P2 - P1" model and came from a measure originally proposed by Asch [39]:

$$I(D|FK) = \text{abs}(P(D|FK) - P(D)).$$

In this model, the measure of uncertainty is simply the probability of the disease (D). The implementation of this model was:

$$I(D|F) = \max \{ \text{abs} (P(D|F^+) - P(D)), \text{abs} (P(D|F^-) - P(D) \}. $$

A fourth and final measure of information gain tested by Guo was called the "weight of evidence" model. In this model, the contribution of a finding FK to the diagnosis of a specific disease versus the alternative, which is the absence of the disease, is given by:

$$W(D|FK) = \log_2 [ P(FK|D) / P(FK|\neg D) ].$$

Guo created five versions of the Iliad best information algorithm using each of these models of information gain and tested them by comparing their sequential workup decisions to those made by a group of expert internists. In general, the Shannon and P2-P1 algorithms were chosen as best by a panel of expert clinicians.

In a complimentary study, the Iliad best information algorithm was rewritten to assess the utility of findings by two fundamentally different strategies. In the first strategy, the best information content was determined \textit{within} each disease
frame in the differential. In this strategy, the utility of each finding is calculated once for each disease and the finding with the highest disease-specific utility is chosen as the "best next information". This is the strategy currently used by Iliad. In a second strategy, the utility of each finding as it related to each disease was summed to give an overall utility across diseases. The hypothesized advantage of this second "across-frame" strategy is that it does not neglect the information contribution of findings related to all but one disease as does the current version which can be said to pursue a "winner takes all" strategy. This study confirmed the results of the previous study, i.e. that the Shannon and P2-P1 strategies were superior, and demonstrated that the "across-frame" strategy was in fact superior overall.

While the current version of Iliad uses the LR model/single-frame strategy, it is clear that future versions may be improved by changing to either the Shannon or P2-P1 model of utility and assessing information gain across diseases. The advantage of the current LR model is that its parameters are easily accessible from the Iliad knowledge base [31]. The main drawback of the Shannon and other models is that they are computationally intensive, requiring the derivation of the posterior probability of each hypothesis under consideration given each possible medical finding. Up until the present, the computational complexity of these algorithms has been a significant limiting feature. However, future implementations of Iliad may run on more powerful hardware configurations with a diminishment in the cost of computational complexity [40] making such techniques more feasible.

Running Iliad

Consultation Mode

The user begins by entering a patient's name, age, sex, and chief complaint. The chief complaint can be entered as free text or selected from an indexed list of findings. Entering a general term such as "headache" will bring up a window requesting more specific information such as the timing and quality of the finding. Next, the program generates a list of possible findings. The user then indicates whether each of these findings is either present or absent.
At this point, Iliad generates its differential diagnosis in order of decreasing likelihood. The user has a number of options. One option is to enter more findings in order to narrow the differential. Another option, which is unique to Iliad, is to request the next most useful information. Iliad uses the utility function described above to display the finding that would most cost-effectively narrow its differential diagnosis. This function utilizes the likelihood ratio and cost data associated with each of the 11,900 disease manifestations to decide on the most useful piece of information not yet known to Iliad[33]. Thus, relatively inexpensive information like history and physical exam findings are requested before costly tests. Finally, the user can request an explanation of any diagnosis on Iliad's working list. The explanation consists of a list of findings and the Bayesian statistics associated with each one as it relates to the disease in consideration.

**Critiquing Mode**

In critiquing mode the physician enters his diagnosis. Iliad requests only the findings to confirm or rule out the hypothesis. As the physician enters these findings, Iliad displays the posterior probability of the disease. The user can use this mode to pursue a diagnostic possibility while saving the list of findings as they are entered. At any time these findings can be reassessed in consulting mode thus giving the user some flexibility in his approach to computer-aided diagnosis.

**Simulation**

In simulation mode, the user is presented with a brief history and chief complaint. He must specify a differential diagnosis and decide what historical, physical, and laboratory findings to request next. Iliad scores the user by generating a number of variables: *Final Diagnostic Errors* is an assessment of the correctness of final diagnostic hypothesis. The *Posterior Probability* assesses the completeness of the workup. The *Average Hypothesis Score* is the sum of the differences between the cumulative utility of each piece of information requested to that of the best information available on each round of questioning. Iliad also keeps track of the *Cost* of the workup and compares it to that of an ideal workup.
Simulated cases can be entered in several ways. A real case can be saved from consulting mode or critiquing mode by selecting a "save as..." command. Iliad can also automatically generate unique simulations of any of the diseases in its knowledge base by applying a statistical filter to its list of findings. In this manner, multiple presentations of the same disease can be generated.

Evaluating Iliad

In 1991 Lincoln and colleagues [30] tested the ability of Iliad to teach medical students. One hundred junior medical students in a six week internal medicine clerkship were randomly assigned to receive Iliad training on two common or uncommon diseases. Each student was subsequently tested on two diseases, one of those they had been trained in and a third disease that they had never seen on Iliad. Dependent measures included each student's Total Diagnostic Errors, the Posterior Probability, and the Average Hypothesis Score for each session. The authors found that for uncommon diseases, students who were trained committed fewer Final Diagnostic Errors and attained higher Posterior Probabilities than students who were untrained indicating that Iliad training improves subsequent performance on Iliad simulated cases. The authors acknowledge the fact that this may not necessarily imply that students will perform better on actual patients when trained by Iliad to diagnose their disease and plan to perform studies in the future to assess this. The authors also suggest some ways of incorporating Iliad into an educational program. Essentially, they suggest that institutions obtain strong faculty support for Iliad-based training, place computers where students can use them conveniently, train students in the use of Iliad, and make Iliad a required part of the training experience.

Conclusions

Why Use Expert Systems? A Case Study

The following case demonstrates the need for diagnostic support for physicians. A young woman presented to her managed care physician with a two-month history of worsening sinus pain, purulent nasal discharge, and weight loss. The physician who saw her worked her up for allergic rhinitis but could not relieve her symptoms. Over the course of several months this woman's physician
referred her to various subspecialists. Each specialist ruled out problems within
his or her domain but none could come up with a satisfactory diagnosis to
account for her symptoms which, by this time, seemed to be worsening and had
come to include weight loss on the order of 20 percent of her body mass. One
day after receiving multiple injections of common allergens as a diagnostic test,
the woman presented with rapidly progressive renal failure, disseminated
intravascular coagulation, and adult respiratory distress syndrome. She died
within a few days of a disease most of her physicians had heard of but had
never seen, Wegener's granulomatosis.

Retrospectively, the diagnosis of Wegener's granulomatosis was not difficult to
make. The full blown clinical syndrome is easily recognized. But on
presentation, this disease can look quite benign. Wegener's is the kind of
disease that can catch physicians by surprise. It is rare, deadly (95% die if
untreated) and treatable (most will survive if treated with a combination of
corticosteroids and cyclophosphamide). We know that physicians learn by
experience. Given that most physicians will never see a case of Wegener's
granulomatosis in their entire career, it is understandable that this disease went
undiagnosed. But is it forgiveable?

In hindsight we might wonder why no one bothered to look in a textbook. It
seems that despite the subacute nature of the problem, no one bothered to stop
and think carefully and come up with a complete differential diagnosis. The fact
of the matter is that physicians tend to be "lumpers" even as they are taught to
split\(^1\). As dangerous as this strategy may seem, the fact of the matter is that it is
reinforced by the mere fact that in most cases it works. The old saying that
"common things occur commonly" shows how much emphasis physicians place
on the prevalence of diseases even when uncommon diagnoses should be in
their differential.

\(^1\) Lumpers consider one or a few diseases at a time while splitters write out a complete differential
diagnosis for each case.
Human Error: It Happens

There are some compelling reasons for incorporating artificial intelligence into the diagnostic routines of physicians. Computers are of tremendous value in the medical domain because of specific failings of the human mind [25]:

_Rhetoric distorts logic._ Physicians are influenced by passionate speech, the manner in which a patient is presented to them, or specific aspects of a case that are emotionally charged for them. This is similar to the argument Spock often had with Captain Kirk on Star Trek. Spock would be faced with a problem and would attempt to make a purely logical decision. Captain Kirk would usually intercede in a wholly illogical manner which would succeed anyway proving once and for all, or at least until the next episode, that the human spirit, with all of its failings, is superior to any purely logical agent. Here we are faced with the prospect that Spock would have made a better diagnostician than Kirk. On the other hand, diagnosis is only a small fraction of what human physicians actually do. Weighing the costs and benefits of various procedures and treatments is a task that will probably require human judgment for a long time (if not forever) since it takes into consideration such things as the patient’s quality of life, his or her ability to withstand intervention, and his or her basic philosophy and outlook on life. Obviously, medicine requires both logic and compassion. When logic is distorted by rhetoric and passion, then emotional distance is a distinct advantage. When logic fails, the physician should be able to draw upon his or her understanding of the human condition. The clinical decision aid should ground the physician in factual data but allow him to make the ultimate decision in the context of the whole patient. This duality should be kept in mind as we consider the role of expert systems in medical practice. Are expert systems an attempt to reduce medicine to its logical essence at the expense of its humanistic ideals? Or is it possible to use expert systems to free the physician from the limitations of memory and logic to allow him to focus on those aspects of medicine which are most human?

_Preconception distorts belief._ Physicians know diseases as they have seen them. Thus, a physician’s assessment of the probability of a disease he has never seen before may be biased toward zero (as in the case of Wegener’s). Moreover, the association of a specific sign with a disease may lead the physician to draw false conclusions and to not consider certain diagnostic
possibilities that are important but less likely. For example, the patient who presents with intense vomiting associated with cholelithiasis may teach the physician that vomiting is an important manifestation of that disease. However, this preconception could very well be a hindrance if the physician does not remember that there are indeed occasions when such patients do not present with vomiting. Expert systems and systems such as Iliad which are designed to teach, have the potential to deliver knowledge that goes far beyond the experience of the average practitioner. An expert system can "remember" to include Wegener's in its differential, regardless of how frequently the disease is encountered. Moreover, a physician using an expert system can draw on the experience of hundreds or even thousands of physicians over many years of practice.

Overconfidence. Physicians tend to be overconfident of their beliefs relative to their knowledge. It is almost as if the act of making a diagnosis somehow increases our confidence in that decision and makes us more likely to see evidence that supports it and to ignore evidence that contradicts it. There is a well described phenomenon called cognitive dissonance in social psychology which may explain this wherein the human mind tends to filter for information that supports its beliefs, rejecting that which does not. Thus, the proverbial diagnostic "blinders" may be more human nature than medical artifact.

Are We There Yet?

In 1994, The New England Journal of Medicine issued its own report card for computer-assisted diagnosis [41] and gave it a "C". Citing a comparison of four commercially available diagnostic computer programs [42], the author complained that the correct diagnosis appeared only 50-75% of the time and that relevant diagnoses appeared just over one-third of the time. He characterized the last forty years of research in medical expert systems as a series of experiments that were abandoned when the real scope of the problem was appreciated. Only a handful of systems have stood up to the demands of real clinical usefulness: that a system performs on par with an expert clinician, has features making it acceptable to clinicians, and encompasses a wide enough range of diseases to be useful. The author had used several of the programs personally and was impressed when they suggested plausible diagnoses he had not considered. However, he was frustrated because too
often such information was buried in a list of implausible diagnoses. To be truly useful, systems will have to diagnose more complicated cases with better accuracy and they will have to allow free text entry of findings as they are encountered, rather than a menu-driven format. He worries that research groups are abandoning the tedious task of expanding their knowledge bases. Most have turned this over to private industry which may or may not continue depending on the financial incentives.

In response, I would assert that all four systems reviewed in that article utilize probabilistic reasoning and that each is characterized by creative compromises for the fact that actual data on the prevalence, specificity, and sensitivity etc. of every medical finding known to man is virtual unavailable. For while medical practitioners seem to be fond of referring casually to such statistics, they rarely use them. Physicians make clinical decisions using a kind of rough probabilistic reasoning [43], have only a vague idea of the influence of a given finding on the likelihood of disease [44], usually consider only a few diagnoses at a time [45, 46], and make extensive use of heuristics and pattern recognition [47]. For medical expert systems, however, this information is essential. It is certainly commendable that the authors of most of the expert systems that are available today have spent considerable time and effort abstracting this information or, more commonly, estimates of these numbers, from the medical literature. However, it seems obvious that the task is truly best performed automatically.

In this respect, medical expert systems may be somewhat premature. While this is definitely an area worth pursuing, it is arguable that a truly successful system will ultimately require accurate statistics on a scale and with precision which can only be accomplished automatically. Therefore, the rate of expert systems evolution may be limited by progress in the domains of clinical information systems, and telecommunications. In fact, one of the most exciting prospects in medical informatics is the possibility that a fusion of expert systems, clinical information systems, and telecommunications may prove to be synergistic as the loop between the gathering and application of clinically relevant data is closed

This notion is supported by the commentary of Edward Shortliffe who recently complained that current medical diagnostic systems do not yet integrate well
with available clinical data systems and are best viewed as "memory joggers" rather than as replacements for human diagnostic skills [48]. Based on his experience with Iliad, Bryan Bergeron [26] would probably agree. He states that one of the biggest problems with Iliad is that it requires the manual entry of patient-specific data into the program and that it is unable to access information stored in a hospital information system. Thus, two of the main goals of medical informatics, reducing the manual and redundant aspects of medical diagnosis and providing access to a large and up-to-date knowledge base, are not currently being met by current systems. It seems only logical that these systems should be integrated, but as yet this goal has not been reached. Medical expert systems remain at the periphery of the informatics revolution. They require a well developed infrastructure that is currently unavailable except at a few forward looking institutions.

Therefore, we are not there yet. The primary technical impediment to the widespread application of expert systems seems to be the fact that they are not yet integrated into hospital information systems (HIS) or the wider medical information infrastructure. As these technologies come on line, medical expert systems will become more powerful and easier to use.

In addition, there are a number of social factors that need to be addressed before expert systems and other medical information technologies can be fully integrated into medical practice. First, physicians attitudes toward medical information technology continue to be negative overall [49]. We need to understand the basis for this attitude and be prepared to improve the image of computers within the medical profession. Second, medical schools have been slow to make medical informatics a required part of their curricula [48, 50-52]. Rapid and successful development of an information infrastructure will require a generation of computer literate medical professionals. Third, medical informatics raises a number of legal and ethical questions that have not been adequately addressed [53-56]. Legislative and policy questions will impede progress unless we anticipate the potential conflicts brought on by these new technologies. Finally, the role of medical informatics is still a matter of conjecture. It is tempting to assume that medical information technology can only improve the quality of medical care and that it will make the practice of medicine more interesting and enjoyable for physicians and nurses. However,
the history of technology suggests that this may be a naive assumption. Real progress is elusive and too often new technologies create more problems than they solve. How will the benefits of medical information technology end up being distributed? In the final section, we will examine the forces that are currently transforming medicine and attempt to reconcile them with the ideals of medical informatics.

Closing the Information Loop

As stated earlier, medical expert systems currently suffer from a number of technical shortcomings. In order to use them, it is typically necessary to re-enter a patient's data directly into the program's database, even if the data already exists in an electronic form elsewhere. Moreover, expert systems typically rely on data whose quality and quantity are currently limited by the fact that they must be abstracted. Both of these problems can be solved by "closing the information loop." When medicine finally incorporates hospital and other clinical information systems on a wide basis and becomes capable of transmitting and sharing its information across telecommunications lines, then expert systems will finally become useful and perhaps even indispensable.

Hospital Information Systems

The capabilities of the typical hospital information system lag far behind those of the most sophisticated HISs in use to day [57]. However, the electronic medical record and the widespread use of hospital information systems is imminent. In 1991, the Institute of Medicine issued a report calling for the replacement of the traditional hospital-based medical record with the electronic patient record [58]. The advantages outlined in that report of having an electronic medical record are numerous. Among them are the following:

The electronic medical record will be more longitudinal than the traditional paper-based record. Information will not degrade over time and will be accessible regardless of its age. Because the electronic medical record will be distributed among the various points of healthcare delivery, it will facilitate a more interdisciplinary approach to healthcare, be more patient-centered and it will encourage less invasive and less expensive care by supporting the
provision of care in multiple settings, including the doctor's office, patient's home, and other healthcare settings.

Another advantage of the electronic medical record is the fact that it will increase the feasibility of outcome studies [59]. Given the economic imperatives of managed care, it is now more important than ever to know the relative cost effectiveness of various therapies and interventions before they are undertaken [60, 61]. Unfortunately, these kinds of studies are expensive and difficult to perform. They require that inputs and outputs be carefully controlled and standardized across settings [62], a feat of enormous magnitude in the absence of computerized records. With computerized medical records, outcome studies will no doubt become standard practice. Every patient encounter will become a chance to collect data for later evaluation in studies of enormous magnitude and power.

For expert systems, the advantages of full featured hospital information systems are obvious. When patient information is captured longitudinally in a standardized, electronic format in, submitting it to an expert system becomes trivial. Moreover, the designers of expert systems will no longer have to formulate their own syntax for clinical information. They will benefit from having a well defined database of clinical findings and patient data to draw upon for inference. It is conceivable that the process of electronically generating a differential diagnosis will one day become commonplace within the context of the computerized medical workplace. Expert system modules may one day become standard equipment on integrated information platforms supporting all aspects of healthcare management and delivery. Reminders and suggestions could be triggered by certain events such as the reporting of laboratory results, the input of new findings, or the submission of a request for certain medications. Accessing a computer generated differential diagnosis may be as easy as clicking on a "view differential" icon.

Another way in which expert systems may one day benefit from the widespread use of hospital information systems and electronic medical records is in widespread use, developers of expert systems will be able to access their databases. By pooling data from multiple sources, large collections of standardized, longitudinal data will be easily incorporated into expert system knowledge bases. Inference could then be based on hard data instead of
estimates. The assembly and maintenance of knowledge bases could be automated. Surely, expert systems research and development would take off under these circumstances.

The logical extension of the hospital information system is a national or international healthcare information infrastructure based on the emerging worldwide telecommunications platform commonly referred to as the internet [63].

Building a Healthcare Information Infrastructure

In 1992, the Association of American Medical Colleges (AAMC) released a publication entitled "Academic Information in the Academic Health Sciences Center: Roles for the Library in Information Management [64]." In that paper, the AAMC made two recommendations: (1) That information management networks be integrated to allow the sharing of academic, operational, and clinical information. (2) That institutions develop academic programs to teach information technology to health care students and faculty. In response to these recommendations, the National Library of Medicine (NLM) created the Integrated Advanced Information Management Systems (IAIMS) initiative. In order to stimulate biomedical academic centers to make their resources available over networks, the NLM provides grant assistance to medical centers and health science institutions and organizations for planning and operational projects leading to the implementation of Integrated Advanced Information Management Systems (IAIMS). These are institution-wide computer networks that link and relate library systems, individual and institutional databases, and information files, within and external to the institution, for patient care, research, education, and administration. The goal of the IAIMS project is to provide systems of comprehensive information access. The advantages of such an arrangement is obvious. By creating a network of biomedical academic information services, their wealth of medical information can be made available to anyone with a computer and a modem.

Since 1983 when the NLM initiated the IAIMS project, seventeen institutions and organizations have received awards for one or more phases of activity. These are listed in table 7.
Gradually, medical institutions around the country have established internet connections. Now, any hospital can get on the internet [65] and access large amounts of data from institutions involved in the IAIMS project as well as several other national medical institutions including the National Institute of Health, the National Library of Medicine, the Health Care Financing Administration, and the center for Disease Prevention and Control [66, 67]. The internet is useful in many settings. Rural providers can use it to access information heretofore unavailable outside of urban academic medical centers [68]. State and regional healthcare networks can use the internet to create their own networked information systems [69]. Moreover, as technology for electronic data interchange becomes more sophisticated, the electronic medical record will gradually become a distributed resource, available to all authorized persons with access to the internet [70].

One day it may be possible to access up to date literature at the point of care [71-74]. By linking together electronic medical records, expert systems, and literature search engines, full-text patient specific resources may become accessible at the bedside. By extending these links to research institutions, peer review boards, and the designers of medical information technologies, the information loop may eventually be closed.
Beyond Problem Knowledge Coupling

The closing of the medical information loop through computers and telecommunications technology is an extension of the concept of problem knowledge coupling, proposed by Dr. Larry Weed, developer of the problem oriented medical record over 20 years ago. In *Knowledge Coupling* [75], Weed predicts a transformation in the paradigm of medical information wherein the healthcare provider will be freed from the limitations of memory and allowed to consider all of the possible causes of identified symptoms and physical findings. Clinical judgements will be left to the physician who is aided by expert systems and all of the resources available over distributed networks. In our wider model, information will flow in both directions. The distinction between clinical trials and everyday care delivery will no longer be relevant (Figure 4). Information will flow from the clinic, to the research institution, to various peer reviewers, policy makers, and knowledge engineers, and back to the clinic in a closed loop that benefits everyone.

These technical advances seem almost certain to become reality in the not too distant future. In many ways this is something to celebrate as we are long overdue for a revolution in the ways in which medical information is managed. However, regardless of their technological benefits, the new information paradigm will present a number of social challenges that must be addressed if we are to succeed. The following sections address these challenges.

Attitudes Toward Medical Information Technology

A number of studies [49, 54, 76-90] have measured physicians' attitudes toward computes in the medical workplace. These studies are summarized in table 8. In one of these studies [49], the researchers were so disappointed in their findings, the author reported the fact, stating that they were "surprised and disheartened by the apparent lack of interest (and occasional hostility) regarding the professional use of computers."
<table>
<thead>
<tr>
<th>Study, Year</th>
<th>Group</th>
<th>Findings</th>
</tr>
</thead>
<tbody>
<tr>
<td>al-Hajjij, 1992</td>
<td>Medical Students and Physicians</td>
<td>Sixty percent had no computer experience. Attitudes were generally negative although most respondents had a positive attitude toward computer education in the medical curriculum.</td>
</tr>
<tr>
<td>Burkes, 1991</td>
<td>Nurses</td>
<td>Satisfaction with computer use positively correlated with computer knowledge, negatively correlated with computer experience and overall nursing experience.</td>
</tr>
<tr>
<td>Detmer, 1994</td>
<td>Academic physicians</td>
<td>Overall, computers were rated as being &quot;slightly beneficial.&quot; Best uses perceived to be self-education and access to up-to-date information. Major concerns were privacy and the doctor patient relationship. Prior computer training was correlated with favorable attitudes.</td>
</tr>
<tr>
<td>Gardner, 1994</td>
<td>Physicians and nurses using the HELP System</td>
<td>Access to patient data and clinical alerts were rated highly. Expert systems were not perceived as a potential source of external monitoring. Patient privacy was not perceived as being threatened.</td>
</tr>
<tr>
<td>Hebert, 1994</td>
<td>Nurses</td>
<td>Intent to use is determined by perceived relative advantage, compatibility with previous work patterns, demonstrability of results, and influence of senior policymaker.</td>
</tr>
<tr>
<td>Mendell, 1991</td>
<td>Administrators</td>
<td>Attitudes were generally positive and correlated with gender, number of employees, perceptions of inefficiency, time-consuming personnel problems, need for efficiency, and institutional climate.</td>
</tr>
<tr>
<td>Murphy, 1994</td>
<td>Nurses</td>
<td>Attitudes became more negative after actual experience with a patient care information system.</td>
</tr>
<tr>
<td>Oliveira, 1994</td>
<td>Medical Students</td>
<td>Self ratings of computer literacy: 63%</td>
</tr>
<tr>
<td>Shortliffe, 1989</td>
<td>Physicians</td>
<td>Attitudes were generally negative.</td>
</tr>
<tr>
<td>Shumway, 1990</td>
<td>Pharmacists, nurses, and physicians</td>
<td>Nurses and physicians knew less than pharmacists about information systems and were more skeptical about their role in reducing costs and improving the quality of care.</td>
</tr>
<tr>
<td>Szecsenyi, 1992</td>
<td>Physicians</td>
<td>Overall positive attitudes except for expert systems that determine referral eligibility.</td>
</tr>
<tr>
<td>Weir, 1994</td>
<td>Physicians</td>
<td>Successful implementation of an order entry system was correlated with available functionality, physician involvement, and the support and involvement of administrators.</td>
</tr>
</tbody>
</table>
Several of these studies report overall negative attitudes toward medical computing among health care professionals [49, 82, 83, 85, 88]. In Table 9, I have listed the most important negative attitudes revealed in these studies.

Table 9. A Summary of Physicians' Negative Attitudes Toward Clinical Decision Support

<table>
<thead>
<tr>
<th>Attitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computers will distance physicians from patients</td>
</tr>
<tr>
<td>Medicine is a distinctly human enterprise that cannot be performed by a machine</td>
</tr>
<tr>
<td>The current generation of physicians has not been adequately prepared to use computers</td>
</tr>
<tr>
<td>Advice from a computer may create difficult legal dilemmas</td>
</tr>
<tr>
<td>Computers will tell physicians how to practice medicine</td>
</tr>
</tbody>
</table>

Many physicians are concerned that the computer can only come between them and their patients. They value the fact that they are the ultimate instruments of healing and fear any mediation of the interaction between themselves and their patients. Those invoking this argument often complain of having to spend extra time and energy at a computer terminal which ultimately reduces patient contact. Indeed, in many cases, this may be a valid concern as when information must be re-entered into stand-alone systems in the absence of an integrated HIS. A related concern is that computers may negatively influence patients' views of their health care encounters. Interestingly, this particular fear may have little foundation. In a study of physicians' and patients' attitudes toward a rheumatologic expert system [91], researchers found that only 10% of patients thought the system negatively affected their relationship with their physicians. Older patients saw greater problems than younger ones. Despite these figures, 82% of physicians feared that the system might undermine their patient's confidence in their abilities.

Indeed, this is only one study and to date there is little data on how computers might impact the doctor-patient relationship. However, we should certainly remain aware of the potential for harm in this area. It is important to create interfaces that compliment the traditional doctor-patient interaction and which do not create more work for physicians. Once physicians realize true benefits from using computers for routine tasks, they will begin to accept the possibility that they may ultimately revolutionize the practice of medicine.

Fears that medical decision-making systems threaten to interfere with the "human" element in healthcare delivery may be extreme [92]. It has been said
that "medicine is . . . both science and narrative, both reason and intuition" [93]. Ideally, computers would not interfere with the doctor-patient relationship. They would enhance it by freeing the physician of the laborious and time-consuming tasks of documenting and accessing information. Success in this area depends on success in other, more technical areas as well as on the purposes to which medical information technologies are applied. If we build humanism into our goals, then computers may serve to enhance the human aspects of medical practice. On the other hand, if we focus only on economic goals, humanism may suffer.

The remaining problems that physicians have with medical information technology are addressed in the remaining sections on computer literacy, legal and ethical issues, and politics.

Of course, not everyone shares these concerns and a few physicians are even enthusiastic. A survey [80] of doctors in Germany, for example, showed that physicians in that country were generally positive about most aspects of medical computing. In that study, younger physicians who were male and had experience using computers in their practices accepted medical computing at the highest rate. However, the tendency was to accept applications that offered access to information but to reject those which made decisions for physicians. Acceptance was highest for systems offering access to databases of drug information and medical literature. An expert system that determined patient eligibility for referral to a specialist was perceived negatively overall suggesting that concerns about the influence of computers on autonomy are international. Another study [90] showed that medical professionals who currently practice in a state of the art informatics environment (LDS Hospital, Salt Lake City, Utah) are satisfied with the role of computers in medical care. In that study access to patient data and clinical alerts were rated highly. Respondents did not feel that expert computer systems would lead to external monitoring, or that these systems might compromise patient privacy. Moreover, physicians and nurses in that study did not feel that computerized decision support decreased their decision-making power. This is indeed encouraging for it suggests that the greatest fear is fear of the unknown. Once medical information technology is successfully implemented, it tends to win converts readily. In fact, it brings to
mind a maxim proposed by one author in the title of his commentary, "Get the computers right, and physicians will get with the program" [94].

Medical Informatics Public Relations Strategies

Ten years ago, software tools were rarely used by health personnel and were viewed skeptically. However, several factors may bring about a change in physician's attitudes. First, computers are smaller, cheaper, and easier to use. They now often fit on a desktop, cost under $2000, and have intuitive interfaces making them accessible even to those who have never used them before. Moreover, computers have been demystified and brought into the realm of everyday use so that society generally thinks that they are manageable and useful. Another important fact is that health care professionals are increasingly distressed by the sheer volume of information they need to access in order to practice medicine well. As computers are revealed to offer better tools for information gathering in this age of overwhelming resources, physicians may come to see them as part of the solution rather than the problem. Perhaps the most persuasive fact will be that fiscal pressures will continue to encourage the practice of cost effective medicine and force practitioners to carefully consider the utility and reliability of tests, procedures, and therapies. When medical information tools begin to deliver a competitive advantage to those who use them, their day will have arrived. They will become indispensable in the health care marketplace.

Computer Literacy and Education

Future medical professionals will rely increasingly on computer technology. Therefore, it is extremely important to educate young physicians to understand and keep pace with emerging technologies [92]. While some medical schools now formally require that all medical students be "computer literate," [95] medical informatics is rarely a required subject in the health sciences curriculum [48]. According to Shortliffe, "schools must develop or recruit expertise in academic computing and must closely connect those people with curriculum planners and reformers" [48].

Medical educational institutions should focus on a number of important goals. Not every medical student will go into medical informatics. However, each will
surely interact with the technology at some point in his or her career. Therefore, at the very minimum, physicians should be taught the following:

(1) Physicians should know enough about hospital information systems to be able to assist their institutions in deciding which system to purchase. Too often systems are purchased without adequate input from practitioners who are familiar enough with them to understand their potential limitations and pitfalls. Given that the purchase of a hospital information represents a significant investment of both money and time, it is imperative that such decisions be made on an informed basis.

(2) Physicians should have enough knowledge about the field of medical informatics to know what to expect from a state of the art desktop workstation. It is important to have an informed professional workforce that knows

(3) Physicians should understand the important issues in decision science (predictive value of tests, cost-effectiveness, Bayesian analysis) and be able to discuss them with their colleagues and with medical informaticists.

(4) Physicians should understand why many clinical computing systems have failed and how they can help ensure that future systems do not make the same mistakes.

(5) Physicians should know the roles of preferences and probabilities in the approach to a patient's management.

(6) Physicians should know about the common reasoning biases that can confound an optimal approach to diagnosis or treatment.

(7) Physicians should know the characteristics of information flow in the modern health care institution and be able to analyze them critically.

Legal and Ethical Considerations

The advances we have reviewed here offer a number of legal and ethical challenges that will have to be worked out as information technology becomes integrated into medical practice. In particular, we should ask ourselves, when the technology should be used, who should use it, who is liable for damages
resulting from its use, and how we might ensure that its use does not threaten patient confidentiality.

When Should Information Technology Be Used?

Obviously, no information technology is inherently beneficial. We cannot expect to improve patient care or to save time or money simply by implementing an automated system within a medical context. Some tasks are more efficiently accomplished manually and automation is not always risk free. Therefore, it is important to carefully evaluate institutional needs before implementing an automated solution. Otherwise, information technology has a tendency to become a solution in search of a problem or, even worse, the cause of new problems within an institution. Therefore, it is important to state clearly the fundamental goals toward which information technology can be ethically applied.

In general, information technology should only be applied when it improves the quality of care at an acceptable cost in time or money or when it reduces these costs without negatively impacting patient care [96, 97]. Improving the quality of care means increasing diagnostic accuracy, delivering more appropriate therapy, raising the quality of life of patients, improving professional morale, informing patients and medical personnel more effectively, and documenting information more efficiently.

Time and money are easy to measure, but quality of care is not. Moreover, in the current managed care environment, fiscal considerations predominate. How then will we determine whether a particular IT solution is truly beneficial when quality is so difficult to measure? Will there not be more incentive to accomplish measurable goals, perhaps even at the expense of some aspects of quality? Even with substantial progress in our ability to measure quality, there may be insufficient incentive to improve it so long as consumers remain sensitive to price and without access to objective and valid comparisons of the quality of various providers.

Who Should Use Clinical Decision Support Systems?

Clinical decision support may be used by physicians, nurses, medical students, physician's assistants, paramedics, laboratory and office personnel, and
patients. Insurance companies and government agencies may be interested in using these tools to monitor the delivery of care or to verify the appropriateness of therapeutic or diagnostic interventions. None of these situations is without its own set of pitfalls.

For example, the most common scenario involves the physician user of a clinical consulting program who seeks advice or additional information. Certainly the physician is best prepared to understand the nuances of the problem at hand and to interpret the output of the program in the context of patient care. If a physician doesn't agree with the system's advice, he or she may override it and do something more appropriate. However, there is no guarantee that physicians will be competent in the use of such programs. Without adequate training, physicians may enter data incorrectly or make other mistakes that could adversely affect the validity of a program's output. Shouldn't physicians be required to demonstrate competence with clinical decision support programs before they are allowed to use them in practice? Perhaps there should be a certification process. Another possibility is to add medical informatics to the curricular requirements for medical licensure.

Another possibility is the use of clinical decision support by medical personnel to make decisions normally relegated to physicians. Possibilities include nurses, physician's assistants, students, and even administrators. Hospital administrators could be tempted to replace physicians with other professionals who will rely on a computer to help them make decisions. Or they may insert the computer at some point upstream of the physician such as at the desk of an appointment clerk or triage nurse. The danger of blindly following the output of a machine is obvious. Even more insidious is the possibility that by relegating medical decisions to a machine, we may reduce the degree to which each of these medical professionals feels responsible for or is intellectually invested in medical decisions.

That is not to say that there are never situations where non-physicians should have access to clinical decision support. When physicians are readily available, such programs should always be used under the supervision of a licensed medical practitioner. However, in places that are short on physicians such as the rural United States or in developing nations, these tools could very well be of benefit.
Another possibility is that unqualified personnel could use such programs to evaluate physicians' performance. For example, a clerk could compare physicians' decisions to those recommended by an expert system. On the basis of such comparisons, insurance companies might be tempted to rank physicians or to deny certain claims for reimbursement. This is problematic. Medical decisions are far too complex to be evaluated automatically. It may be possible to use such systems as a screening tool, but each case should ultimately be reviewed by a qualified professional. Otherwise, physicians may be subjected to undue criticism or inappropriately denied payment.

Patients may also seek advice from clinical decision programs. This poses some difficulty. Some patients may be tempted to bypass skilled medical professionals in favor of self diagnosis and treatment on the basis of patient specific advice from an expert system. This may seem no more dangerous than having a patient look up his or her own symptoms in a book, but it is Books don't typically offer patient-specific advice. Moreover, books seldom imitate thought whereas computers do. People are far more likely to endow the computer with "intelligence" if only because it seems to "know" something about what it is told. With copayments and other disincentives to seek the care of medical professionals, patients may attempt to use expert systems as a substitute for medical care. Again, this is better than nothing and for some persons, nothing is the alternative. However, it is important to resist giving patients or anyone else the impression that expert systems can be used as a substitute for the care of compassionate and knowledgeable human beings.

The Problem of Liability

The provider of a product or a service is liable for any harm caused by its use. Liability for damages caused by medical products can be imposed on any of the companies or individuals involved in its use including the manufacturer, the wholesale or retail distributor, the hospital, or the person or persons who used the product at the point of care.

Current legal liability is based on the principles of tort law and may be founded on one of three premises, intent, negligence, and strict liability. The likelihood that a medical information technology product would be made to intentionally harm someone is small but if this were to happen, liability would be assigned on
the basis intent. Presently, most tort liability is based on the premise of negligence. In order to collect damages under this premise, the injured party must prove that the provider of a product or service was negligent. Negligent behavior is that which is deemed to be different from that which a reasonably prudent person would practice under similar circumstances in order to avoid injury to another. For physicians, this is the standard for proving malpractice. In order to prove that a physician behaved negligently, one must show that his or her actions diverged from the "standard of care" of the medical profession. The newest and least well defined theory of liability is that of strict liability. Under this theory, it is the nature of the product that is called into question, rather than its quality. Thus, even if a product is designed and manufactured without negligence, a provider can be held strictly liable for injuries related to a product's design.

The negligence theory of liability is applicable to patients injured by actions taken under advice from a medical computer program [55]. The situation is similar to when a physician consults a colleague regarding the care of one of his patients. It is his responsibility to accept or reject the advice of that colleague based on his own professional judgement. Giving or accepting advice that is below the standard of care constitutes negligence and is the basis on which either physician may be held liable.

However, the situation involving consultation with a computer is actually somewhat different. There is currently no legal precedent for establishing liability in such cases, although it is arguable that negligence will hold for both the consulting physician and the manufacturer of the program. However, in order to prove liability for a product, one must show that the product is not only defective but that it is the legal cause of injury. Manufacturers of consultation programs may reasonably argue that their products are meant for use by qualified medical professionals who should be able to evaluate the utility and appropriateness of their output. If the manufacturer can establish that the program was intended to "supplement the physician's medical information-acting as a library or in the place of a consultant- but does not make decisions for the physician," then liability might be mitigated or avoided entirely provided it can be shown that the physician's negligent use of the product had been unforeseeable [98].
There is another twist to the problem of liability and that is the question of whether the use of computer consultation programs and other medical information technology might one day become the standard of care. When the use of such devices can be established as offering substantial benefit, injured parties may argue that their omission from the treatment process amounts to negligence. Physicians may one day be found negligent for not using medical information technology. We may be years away from such a decision, but it is important to realize that the question is inevitable and that its answer may determine a precedent for how medicine is practiced in the future.

The Problem of Confidentiality

The question of confidentiality comes up in the context of the electronically stored medical record. Indeed, confidentiality appears to be patients' greatest concern when it comes to medical information technology. A study of patient's attitudes toward computerized systems for medical interviews found that patients had no problem with the clinical use of the system but feared that their private data might be somehow more vulnerable [87]. In another study 31% of surgical patients expressed fear of abuse of private data. Clearly, these fears are not unwarranted as the electronic medical record offers a number of challenges to the confidentiality of patient information.

The physical nature of the paper medical record may be a problem in many respects, but when it comes to confidentiality, it is a distinct advantage. An attorney fighting to protect a patient's privacy is in a much better position if the medical record is in his office and not on, say, a server accessible over the internet. Unlike digital records, paper files can be locked in a safe, hidden, or even shredded. Given the sensitive nature of medical information, it is sometimes necessary to invoke such measures under the threat of improper or unfair use. Copying paper records is both time consuming and laborious and often produces material that is illegible or otherwise less valuable than the original. Although there are standards for recording medical information in paper records, they are seldom enforced and there is plenty of room for improvisation. Physicians often take advantage of this flexibility and encode information that could be detrimental such as a patient's HIV status. Another option is to simply leave such information out of the medical record or store it in a separate place.
In stark contrast to the paper medical record, the electronic form can be easily intercepted, reproduced, and transmitted by unauthorized individuals or agencies. Moreover, most electronic medical records will no doubt have standardized formats rendering them less capable of storing encoded or idiosyncratic information. Potentially damaging information will have to be stored in the standard format or left out of the record entirely. When electronic information is subpoenaed in a legal case, it may be transmitted to hostile parties long before an appeal can be filed.

Dealing with these challenges will no doubt be the subject of considerable controversy. Legal measures have been proposed by Blake in a review of the problem [56]. There, she argues for the establishment of a number of guidelines for informing patients about the electronic nature of their records and who has access to them, authorization for disclosure, penalties for violation, and requirements for technical safeguards against unauthorized access or misuse. However, such guidelines must be carefully considered in order to prevent making the proper use of electronic medical records unduly cumbersome.

Technical solutions to the problem of maintaining the confidentiality of electronic medical records include encrypting sensitive information before it is transmitted and requiring passwords for accessing it [99-101]. Access should be monitored automatically by software that records the identity and location of the user, the time, and the information accessed for each encounter. Authority and resources should be granted for following up on instances of improper access. By instituting measures such as these, it may not be possible to protect against all unauthorized access. However, it is important to make the effort substantially expensive or difficult and subject to dissuasive penalty.

Politics, Power, and the Art of Electronic Medicine

Government and societal enthusiasm for science and technology combined with the emergence of third party reimbursement drove the health care economy from 1950 until the late 1980s. In The Social Transformation of American Medicine, Paul Star explains how this happened [102].

As third parties, both private insurers and government programs effectively insulate[d] patients and providers from the true cost of treatment decisions and so reduce[d] the incentive to weigh cost carefully against benefits. From 1960 to 1975 the share of health care expenditures paid by third parties increased from 45 to 67
percent. Like most private plans, Medicare and Medicaid reimburse[d] providers on a fee-for-service basis. Since under fee-for-service, doctors and hospitals make more money the more services they provide, they have an incentive to maximize the volume of services. Third-party, fee-for-service payment was the central mechanism of medical inflation.

The results of these inflationary forces are depicted in figure 5.

Figure 5. Medical Inflation 1960-1995 (Source: HCFA data [103])

Healthcare Spending in the United States

The major players in the financing of health care in America are the Health Care Financing Administration of the Federal Government (overseeing Medicare and Medicaid) and those corporations paying for a large measure of health care insurance for working Americans. Throughout the 1970s and early 1980s, these agencies kept pace with medical inflation by passing the costs of health care on to consumers and taxpayers in the form of price and tax increases. However, in the early and mid 1980s, two major forces began to disrupt this cycle. In the political arena, America experienced a conservative revolution.
predicated on the assertion that its citizens were overtaxed by a liberal establishment with a failed socialist agenda. With the election of Ronald Reagan in 1982, this country entered an era of heightened sensitivity to the cost of government fanned by demagoguery as well as legitimate concern over government excess. On the business side, the rise of international competition forced executives to consider costs in a global context. Passing rising health care costs on to consumers became unrealistic as American corporations went head-to-head with their foreign counterparts in Asia, many of whom enjoyed socialized medicine or had not yet achieved the expensive medical standards of the west. In short, the party was over. It was time for the providers of medical care to make their contribution to economic reform.

The Federal Government Gets Tough

Negotiations between providers and payers intensified and led to a number of fundamentally new strategies. The federal government responded by designing a new system of reimbursement enacted in 1983 with passage of the Tax Equity and Fiscal Responsibility Act (TEFRA). Under this new scheme, providers of Medicare services were reimbursed at a prospective rate per case irrespective of their costs. Patients were (and still are) classified under one of several hundred diagnostic related groups (DGRs) and providers were reimbursed on the basis of these groups regardless of how much of their resources were consumed in treating the patient.

Business Goes Shopping

Meanwhile, businesses began to shop around for a less expensive alternative to the traditional fee-for-service arrangement. Insurers responded with new arrangements known as health maintenance organizations (HMOs) and preferred provider organizations (PPOs) which provided comprehensive medical coverage for an annual fee. The fundamental innovation in the private sector was the adoption of this form of payment which came to be known as capitation. The idea was to control costs on the provider side while ensuring a steady stream of income from consumers. By carefully assuming risk and closely monitoring the utilization of resources, these managed care organizations succeeded in carving out a substantial portion of the health care market. Unfortunately, health care costs continued to spiral out of control.
Flirting With Reform

A debate on the efficacy of managed care and capitation was initiated by reform-minded academics and public interest lobbyists in the early 1990s. Critics of managed care questioned the ability of the private sector to act in the public interest and suggested that its incentives were inherently biased toward the dumping of sick patients, the selection of healthy ones, the withholding of treatment, and the establishment of regional monopolies. Armed with statistics on the rising proportion of uninsured Americans and the relentless escalation of health care spending, they lobbied for a broader role for the federal government in health care administration. Others countered that it was too early to call managed care a failure citing the fact that only a fraction of the American population had been enrolled in capitated programs. They argued that substantial savings would be realized only after a significant proportion of Medicare beneficiaries (consumers of the largest proportion of federal health care dollars) could be enrolled.

In 1989, Douglas Wilder was elected Governor of Virginia becoming the first black governor in American history by running against the health care industry. Following Wilder's lead, Bill Clinton made health care reform the centerpiece of his successful bid for the presidency in 1992. On September 22, 1993, Clinton unveiled his Health Security Act in a speech before a joint session of Congress. Characterizing the American health care system as "badly broken," Clinton outlined a plan wherein the federal government would regulate the insurance industry in the manner of a public utility and mandate the provision of health insurance by employers. Despite its obvious albeit feeble capitulation to the insurance industry, lobbyists and conservative politicians attacked the bill, casting it as a bewildering 5000 page rough draft for regulatory disaster. A campaign style bus tour promoting the plan was no match for the slick television ad campaign mounted by the insurance industry and after much strategic delay, the bill languished on the floors of both houses before being led "into a ditch" by Republican leadership. A few months later, California voters would reject a Single Payer Initiative by a wide margin making 1994 the year in which Americans said "no" to socialized medicine.
The Corporate Revolution

Ironically, the back-to-back failure of both the Health Security Act and its expectant California cousin may also be viewed as the beginning of a revolution in health care financing. By rejecting a broader role for government in the administration of health care financing, Americans have implicitly entrusted big business with the task of reforming its ailing health care system. Thus, we are witnessing a corporate revolution wherein those with the right knowledge and sufficient capital are now free to exploit the American health care crisis and turn it into an opportunity to earn substantial profits. This turn of events has angered some and pleased many [104], but regardless of one's political leanings, we must now face up to the fact that the future of American health care is in the hands of a consortium of corporate agencies whose primary responsibility is to stockholders. Corporate medicine is on the horizon.

Phases One and Two

In The Social Transformation of American Medicine, Paul Starr predicts that the rise of corporate medicine will occur in two phases much like that of other sectors of the American economy during the mid-twentieth century. Phase one will be dominated by "horizontal integration" wherein the nation's standalone health care institutions will be merged into regional networks of providers. The primary advantages incurred during this phase will result from the ability of such networks to "corner the market" and effectively control price and output. In stage two, these networks will be "vertically integrated" into larger, centrally managed conglomerates that will increase profits by developing economies of scale. In fact, since the publication of The Social Transformation in 1982, Starr's predictions have proven to be remarkably accurate. Regional networks currently dominate health care delivery in areas of substantial population density and a few major conglomerates now appear to be in a position to acquire these networks on a national scale. Thus, by Starr's paradigm, we have progressed slightly more than half-way along the trajectory toward a complete corporate "revolution" in health care delivery.
Phase Three

However, if the history of corporate America continues to be a reliable guide to the future of health care delivery (as it has proven to be by forshadowing events in the health care sector by ten to fifteen years), then all signs suggest that the revolution will continue. The next phase seems likely to occur between now and the middle of the twenty-first century and will be characterized by a phenomenon which Starr could not anticipate in 1982, namely the "impllosion" of the modern corporation that results from the adoption of advanced techniques in computing and telecommunications. The non-health related sectors of the economy have only just entered this phase wherein the ideals of a "global village" seem to have captured the imagination of consumers and providers alike. The excitement this "cyber" revolution has generated is manifest in the media, on university campuses, and in the business community and seems to be related to the fact that it is as much a social revolution as a technological one. At the very moment that technology seemed capable of alienating us once and for all, it suddenly promises to unite us into a pure democracy of infosurfing, home shopping, technocitizens. All this in less than a decade. Phase three is beginning.

Medicine seems likely to ease into phase three on schedule, beginning perhaps in the next decade. The obvious advantage of this delay is that medicine can learn from the mistakes of its predecessors. Moreover, as corporate medicine completes phases one and two, we are presented with the opportunity to experiment on various scales before committing ourselves to any particular paradigm. In short, medicine is at a turning point.

The IT Medium and the Message of Managed Care

Dr. Peter Gardner had been practicing surgery since 1965. He knew surgery, he had excellent judgement, and he loved his patients. You could tell by the way he lectured. He brought each case to life. He told me that the practice of medicine had been a great privilege, but that he now looked forward to his retirement. "Medicine is changing," he said, "and I fear that you may never be able to love the profession as I have loved it."
He told me about a patient he had been called to see in the emergency room, a woman he had seen a few days earlier about how to deal with the fact that she had been diagnosed with an aortic aneurism. They had talked at length. She was 85 years old and in excellent health. Surgery was an option, but it was risky. She could develop complications or even die. She would also have to stay in the hospital for several days. That too was risky, especially at her age. He then told her that her aneurism was relatively small and that it might never even bother her. He would leave the decision to her but he emphasized the fact that surgery would definitely interrupt her life. They talked about her children and how nice it was to live near them, especially now that she had a new grand daughter. After a while, she made her decision. No surgery. Her days were too precious to waste in the hospital.

Now she was back. In less then one week her aneurism had enlarged dramatically. Her life was in immediate danger. The physician was a bit surprised. He had played the odds and lost. He smiled at the woman. Fortunately she smiled back. "Remember that procedure I was telling you about? Well, it isn't all that dangerous. You're going to be alight."

He followed her to the OR and assembled a team. He was scrubbing for surgery when someone held a telephone to his ear and said "her insurance company wants to talk to you." The operator (or whatever he or she was) explained that the woman's policy did not cover aortic grafts or any other abdominal surgical procedures for that matter. She was too old. The cutoff for aortic grafts was 82.

The doctor was furious. The operator was polite but firm. "I'm sorry but she did sign the policy. I believe the cutoff is based on a study. No, I don't know where it was published. You're right, she doesn't sound like your average 85-year-old. Granddaughter... Yes, I understand. Look, no one is telling you how to practice medicine. I'm just telling you what it says here in this book, no aortic grafts after age 82, that's all I know."

This story demonstrates how managed care threatens the art of medicine. It is one thing to create guidelines. But guidelines must be applied in light of sound clinical judgement. Someday, managed care organizations may replace telephone operators like this one with information systems that guide
therapeutic decisions in a "cost-effective" manner. The question is, how do we prevent situations like this one. My point is captured in an important paper by Randolph Miller [105] which states that the standard view continues to be standard: people, not machines, understand patients' problems. If the values of managed care are built into medical information technology, they may come to dominate the practice of medicine at the expense of its highest ideals.

Now, I believe that the medium of medical practice conveys a message. Values are manifest in the infrastructure of medicine. To leave the development of medical information technology in the hands of corporate medicine is to entrust the future of medicine to market forces. Granted, there are plenty of ethical and responsible medical professionals working in medical informatics right now. However, I would argue that these individuals are at a disadvantage. Those who will purchase medical information technology will call the shots. Therein lies our dilemma. If medical corporations constitute the market for medical information technology, how do we resolve the aforementioned conflict of interest? Who will advocate the rights of patients? What force will preserve the role of the physician in patient care and prevent the reduction of the medical profession to a cadre of technicians blindly carrying out the instructions of machines designed to maximize the bottom line?

Related questions go beyond the conflict of interest between patients, their physicians, and the new medical corporations. For example, what exactly do we want out of new information technology? Is the goal merely to reduce costs? Or will it be possible to do even more? Can we begin to develop a vision of how information technology might be used to make medicine not only more affordable or more profitable, but better as well? If so, what challenges can be anticipated? Will it be sufficient to regulate medical information technology in the manner of other medical devices? How can we promote the art of electronic medicine?

I propose that we should come up with a set of principles designed to guide our assessment of medical information systems. These guidelines should be based on fundamental values derived from humanism and from the better aspects of the western medical tradition, namely the values of compassion, dignity, privacy, and equity. By taking up such a debate, we might begin to develop consensus and thereby gain some measure of political or professional
influence as practitioners and promoters of the art of medicine under the paradigm of medical informatics. The point is, the rational development of new technology must be guided by principle. We should promote our highest ideals and be wary of baser motives. For ultimately, the role of information technology in the future of medical care will be determined by a synthesis of ideas, rooted in values great and small. The goal is to develop a better appreciation for the current pace of medical information technology, to understand the range of its potential, both good and bad, and to prepare ourselves for the challenge of shepherding the new paradigm of medical information into the twenty-first century.

This thesis is written for my peers who understand that medicine is changing but remain idealistic. They know that medicine is a boat without a destination, but view themselves as its new captains. Surely they will find their way. But medicine is also a boat being completely overhauled. If we plan on taking her anywhere in particular, we should pay close attention to how she is being rebuilt. We should tell the builders exactly where we want her to go or, better yet, grab a hammer and lend a hand. Otherwise, medicine might end up taking us no further than the bank. And that isn't very far at all.
REFERENCES


58. Dick, R.S., *The computer-based patient record: an essential technology for health care 1991*, Committee on Improving the Patient Record, Division of the Health Care Services, Institute of Medicine:


94. Gardner, E., Get the computers right, and physicians will get with the program. Mod Healthc, 1993. 23(9): p. 35, 38-40.


