6/1/84

Heuristic Programming Project
Report No. HPP-84-29

June 1984

Inferring an Expert's Reasoning by Watching

David C. Wilkins, Bruce G. Buchanan and William J. Clancey

Department of Computer Science
Stanford Univeristy
Stanford, CA 94305

Wilkins, D., Buchanan, B. G., Clancey, W. J. (1984). Inferring an expert's reasoning by watching. Proceedings of the 1984 Conference on Intelligent Systems and Machines (pp. 51–58).

Inferring an Expert's Reasoning by Watching

David C. Wilkins, Bruce G. Buchanan and William J. Clancey

Computer Science Department
Stanford University
Stanford, CA 94305

Abstract

This paper introduces a system to infer automatically the model of an expert medical diagnostician by watching the expert diagnose a patient. Our approach relies heavily on a close correspondence between the system and a human with respect to knowledge organization, inference methods and discourse language. The described system is a major component of a *learning by watching* system being created to facilitate automatic acquisition of new problem solving knowledge and to refine cognitive models of expert reasoning.

In this paper we describe the closed world learning environment that allows an expert to solve a problem and explain his or her reasoning to the system. Each time the expert requests data, a hypothesis generator produces a set of alternative interpretations of the behavior. These are pruned by a hypothesis evaluator currently under construction.

1 Introduction

Artificial intelligence has been successful in producing problem solving systems that demonstrate expertise in limited domain areas within fields such as symbolic integration, medical diagnosis and organic chemistry [LIN80]. A bottleneck in the creation and expansion of these knowledge-intensive systems is knowledge acquisition. Acquiring the necessary domain knowledge is a very tedious and time-consuming manual process requiring many person-years of effort on the part of a domain expert and a knowledge engineer. There is good motivation to automate this process, but methods to date have proved unsuccessful.

We are taking a *natural systems* approach [HOL75] to this problem. This means that we are trying to create a framework whereby an expert problem solver's knowledge organization and knowledge acquisition methods are modeled as similarly as possible to human problem solvers.

There are a number of reasons why medical diagnosis is an attractive domain in which to explore this approach. Foremost is the existence of a large body of psychological studies on how human clinical expertise is organized and used [ELS78, KAS78, PAT81, POP82]. A number of medical expert systems have been constructed that approximate the knowledge organization and problem solving method suggested by these studies. In contrast to systems such as Mycin, the creators of the Neomycin, Pip, and Internist medical expert systems view their programs as simulations of the process of clinical reasoning [CLA81, PAU76, POP82].

Learning by observing seasoned experts is a very important step in the development of medical expertise. Prior to observing experienced physicians, a medical student first spends two or three years studying and acquiring textbook knowledge of diseases and the physiology of the human body. At the end of this period, despite a significant repertoire of factual medical knowledge, the student is unable to demonstrate any real diagnostic expertise. Since we define medical expertise as that body of knowledge built

up over textbook knowledge, this comes as no surprise. Expertise is acquired during an apprenticeship period in which the student watches his or her mentors diagnosing real cases and attempts to duplicate this skill on his or her own through practice [KAS82]. The novice's knowledge base undergoes a reorganization during this period.

Our system watches a physician-patient dialogue and tries to determine the physician's reasons for each of the questions asked of the patient. When a question cannot be explained, our system assumes that the expert possesses some knowledge that it does not, and then tries to acquire this knowledge.

The ability to infer the reasons for the action of another expert when watching the expert solve a problem is as much a dimension of expertise as problem solving, explanation of expertise, and teaching of expertise. A familiar example of this within the field of artificial intelligence is seen during organized human vs. machine chess matches. There is often a highly ranked player present who explains the probable reason for the moves of each player during the game. Medical diagnosis is another domain where experts in the same area of specialization are very good at critiquing each other's problem solving behavior. When a physician asks a question of a patient, another physician watching the patient-physician interview can usually infer the reason for each question asked of the patient.

2 The Simulation of Clinical Expertise

The major results on modeling clinical expertise relate to the way knowledge is chunked in long term memory, the relationship between domain and strategy knowledge, and the role of short term memory in managing focus of attention and problem solving. This section reviews three distinguishing characteristics of medical expert systems designed to simulate clinical reasoning, and then describes the ways our research will contribute to refinement of existing models of clinical expertise.

The first characteristic of these systems is representation of the medical knowledge as a network of frames centered around diseases, clinical states and findings. Examples of slots in these frames are findings (observations including facts from the the patient's history, data about the patient's present illness elicited from the patient), caused-by (diseases or clinical states that can cause this disease or clinical state), cause-of (diseases that are caused by this disease or clinical state), complicated-by, complication-of, triggers (usually a single finding that is very strongly connected with the disease or clinical state, prompting the disease to be placed on the differential), differential-diagnosis (evidence that helps distinguish between two similar and commonly confused diseases), subsumes (disease frames that this disease subsumes), and subsumed-by (disease frames that this disease is subsumed by).

The second distinguishing characteristic of these systems is the use of a hypothesis-directed diagnostic technique organized around a hypothesis list called a *differential.* A differential is the list of hypotheses a physician has in mind as possible causes of the patient's symptoms. Much of the diagnosis involves operations on a differential of hypotheses that represents the system's and physician's short term memory. Examples of operations on the differential are confirming, eliminating, and refining an element, grouping elements together and differentiating between elements or sets of elements, and narrowing and broadening the set of elements. Questions that the system asks have one of three purposes: affecting the state of the differential, clarifying or characterizing a previous question, or asking a routine question such as those covered in a head-to-toe exam [CLA84].

The third characteristic of these systems is the separation of strategy knowledge and medical domain knowledge. Ideally, the strategy knowledge should be domain independent, mak-

ing no mention of medical knowledge. Designing a representation that allows strategy knowledge to be cleanly represented is one of the major goals of the Neomycin medical expert system. Neomycin's strategy language provides a convenient framework for creating new strategy rules.

The current approach used to improve programs that claim to simulate clinical reasoning is principally to see how closely the line of reasoning in consultation sessions mimics that of humans on cases other than those on which the program was tuned. Programs are then modified to reduce deviations between the program and human experts. But even when these changes are made, it is difficult to decide exactly how to change the program so as to more accurately model the underlying process of clinical reasoning. And just because a program seems to do a good job of mimicking an expert does not guarantee the fidelity of the model upon which the program is based.

Our approach is to see if a program based on a model of clinical reasoning provides the necessary constraints to infer the model of the expert by watching. We attempt to infer a physician's model at each point during the consultation session when the physician asks a question of the user. This is exactly opposite to the current approaches whereby a program is run (either in consultation, explanation, or teaching mode) and a physician, psychologist, or computer scientist tries to infer the (reasonableness of the) model of the program. Inferring the model of the expert is, of course, a much more difficult task, and the deficiencies in our model of clinical reasoning should quickly evidence themselves. Individual differences between diagnostic styles of physicians make this a particularly challenging task. Yet we know that the possession of expertise does give a human the ability to infer the model of different physicians, so we should expect a good model to provide us with the basis for doing this.

Each time our program fails to infer the correct user's model, protocol analysis is used to determine why the program was deficient. *Ad hoc* changes to the program to correct specific cases are not allowed. Rather, any modifications involve changes at a high level of generality. For instance, changes to the strategy knowledge are domain-independent, making no reference to medical domain knowledge. Our program shall certainly fail at times because of major deficiencies that are beyond both the frontiers of expert systems and psychological studies on medical expertise. An example of this is inferring the user's model in a situation that requires complex temporal reasoning or requires heavy use of mechanistic models of physiological processes.

3 Experimental Framework

Our research focuses on a phase of clinical reasoning known as taking the *patient's present illness* – the typical consultation a patient has with a general practitioner in which a patient presents a chief complaint. This section describes the method used to gather protocols of this situation and the modeling and learning environment constructed by analysis of the protocols.

Novice and expert problem solvers are presented with a case, initially consisting of name, age, sex, race and initial complaints of the patient. Our physician, Dr. Curt Kapsner, answers questions asked about the patient from the point of view of a medical person who knows the patient's medical history and present illness record. The questions must be specific, and general questions such as "describe the stiff neck are not answered. When a question is asked, the expert also tells why it is asked. The answer is then given. Upon receiving an answer, the expert states what was learned from the information and any hypotheses under consideration.

This data collection methodology was applied to tuberculosis meningitis and subarachnoid hemorrhage cases. Based on a careful examination of these protocols, a "closed world diagnostic problem solving environment

has been constructed. Using menus, the expert requests information on observations which includes the patient's history, present illness and lab data. There are also menus for disorder related hypotheses: diseases and psychopathological states. These menus, in conjunction with a menu for communicating to the system the expert's *Reason For Question* (RFQ), allow a user to diagnose a case and critique his or her reasoning without ever touching the keyboard. All entries are input by selecting graph and menu entries with a Xerox D-Machine mouse button. This lead to a friendly user interface and provides our program with all the information needed to model the expert.

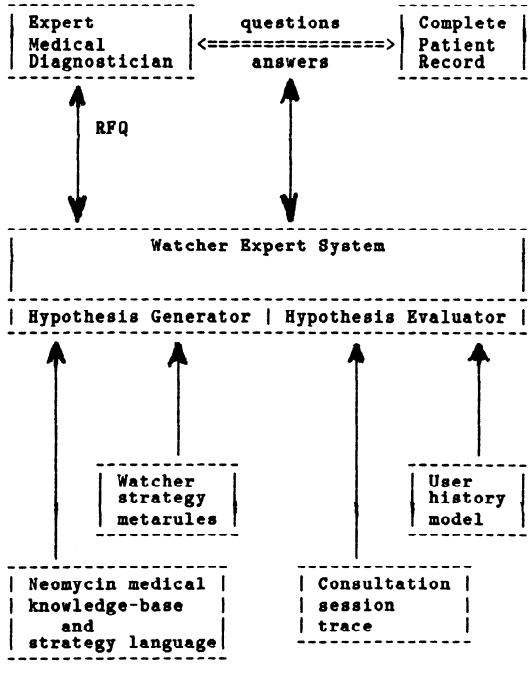

Figure 1: System Organization of Watcher.

4 Hypothesis Generation

Hypothesis generation involves covering the data acquisition strategy space of the expert. For each question the expert asks, the hypothesis generator outputs the various ways the question relates to elements on the differential, routine diagnostic questions, and previous data requests. The Reason-for-Question (RFQ) taxonomy shown in figure 2 provides the kernel of the hypothesis generation routine. Strategy metarules associated with each RFQ task search the knowledge base for alternate RFQ interpretations.

1. Reason-for-question
   - Hypothesis-related
     - Strongly-Confirms
     - Confirms
     - Weakly-Confirms
     - Strongly-Eliminates
     - Eliminates
     - Weakly-Eliminates
     - Discriminates
   - Finding-related
     - Necessary-to-Clarify-Finding
     - Just-Gathering-Info-On-Find
   - General-exploration
     - Review-of-systems
     - Something-always-asked

Figure 2: Reason For Question (RFQ) Taxonomy.

The hypothesis generation metarules begin by performing bottom-up reasoning from the data request to disease hypotheses. This produces a set of hypotheses for which the expert's data request provides evidence. The set is then pruned to include only hypotheses on the differential and hypotheses related to elements of the differential. The data-driven search starts with the specific data item that the user requests and all subtypes of this data item, since the user might be trying to confirm a hypothesis by subsumption. The set of generated hypotheses are presented to the user who chooses the one closest to the reason why the question was asked.

An example of the output of the hypothesis generator is illustrated in figure 3. The expert requests information concerning the observation

"fever" and there are three items are on the differential: tension headache, meningitis, and subarachnoid hemorrhage. The hypothesis generator executes seven strategic tasks represented by the leaves of the RFQ taxonomy (i.e., confirmation, elimination, discrimination, characterize-or-clarify-finding, etc). Three RFQ critiques are returned by strategy task metarules. By searching for explanations in the context of a consultation, the pathways from data to hypotheses are sufficiently constrained so as to allow identification of the correct pathway.

5 Hypothesis Evaluation

Section 4 described generation of a set of candidate hypotheses to explain the expert's reasoning. This section addresses selection of the hypothesis most likely to be the correct interpretation of the observed behavior. As stated earlier, this part of the system is under construction.

Initially, the expert provides an RFQ each time a question is asked. If for each question, the RFQ of the expert is contained in the RFQ list produced by the program, then our hypothesis generator is judged complete. If not, the protocol information will provide guidance on how to refine the hypothesis generator. This process will be repeated using a second observing expert who critiques the first expert's behavior. If we choose our RFQ taxonomy at the right level of abstraction, then there will be strong agreement between the two experts; otherwise our RFQ taxonomy will need to be refined.

The evaluation of competing hypotheses produced by the program can also be done by the expert diagnosing a patient case during the consultation. The hypothesis generator produces the set of possible interpretations of the expert's data request as the data request relates to the differential; and this is displayed to the user in the closed world critiquing language described in section 3. The user chooses the critique that best describes the reason for asking the question.

```
* SHOW-DIFFERENTIAL
>
> tension-headache, meningitis
> and subarachnoid-hemorrhage
>
>
* Does the patient have a FEVER (Y or N)?
>
> RFQ1: fever = yes
> hypothesis related: confirmation,
> strongly-suggests
> meningitis by subsumption
>
> RFQ2: fever = no
> hypothesis related: elimination
> weakly-suggests tension-headache
>
> RFQ3: fever = yes or no
> hypothesis related: discrimination
> yes: meningitis, no: tension-headache
>
>
* What is patient's HEADACHE-CHRONICITY
* (Acute, Subacute or Chronic)?
>
> RFQ1: headache-chronicity = acute
> hypothesis-related: confirmation
> strongly-suggests
> acute-bacterial-meningitis
>
> RFQ2: headache-chronicity = acute
> hypothesis-related: confirmation
> suggests subarachnoid-hemorrhage
>
> RFQ3: headache-chronicity = acute
> hypothesis-related: confirmation
> weakly-suggests viral-meningitis
>
> RFQ4: headache-chronicity = chronic
> hypothesis-related: confirmation
> suggests tension-headache
>
> RFQ5: acute or chronic
> hypothesis-related: discrimination
> acute: meningitis subarachnoid-hemorrhage
> chronic: tension-headache
>
> RFQ6: headache-chronicity
> characterize-or-clarify-finding:
> headache
>
> RFQ7: headache-chronicity
> just-gathering-information-on-finding:
> headache
```

Figure 3: Example of RFQ critiques generated by hypothesis generator.

The next step is to construct an expert system to solve the task of hypothesis evaluation. We will soon begin collecting protocols from experts on their methods of arbitrating between competing hypotheses. This information will provide the empirical data to automate the hy-

pothesis evaluation process. Recall that experts can do this successfully because it is a dimension of expertise. The expert uses a variety of knowledge sources in reaching a decision. Many involve reasoning about patterns of interpretation in the consultation session.

Notice that once again we are involved with a hypothesis formation task. How does an expert form hypotheses regarding which interpretation of another expert is the correct one? The data used to solve this hypothesis formation task is similar to the data an expert uses to solve the medical diagnosis task. We begin with the same data used during the consultation session, but abstract it in different ways. For example, if three questions in a row relate to the same hypothesis, this provides suggestive evidence that the next question will relate to the same hypothesis. Another example is the case where one hypothesis is pursued for a period of time, but abandoned before satisfactorily determining its likelihood. This provides suggestive evidence that it will be returned to, and this pattern should be expected.

The hypothesis evaluation subsystem comprises a complete expert system in itself. This system searches for patterns of interpretation of the user's behavior using the Neomycin knowledge base, a trace of the consultation session, and the user's history. Our embedded expert system that abstracts trace data into categories of interpretation is also a diagnostic or classification expert system like Neomycin [CLA84].

Querying an expert regarding his RFQ is a crutch that we only plan to use at the beginning of the phase of our work on hypothesis evaluation. Experts do not require this crutch, so neither should our program. The RFQ will be determined directly from the consultation session, and the user will only be asked when the system produces several different explanations with high certainty factors. Even then, this might not be necessary since later questions could aid in disambiguating these multiple interpretations.

6 Comparison to Other Learning by Watching Systems

The ability to infer a user's reasoning is an important element of any system to do learning by watching. This section describes some well known systems for learning expertise by watching.

Samuel's checker player was one of the earliest system that acquired expertise by watching a human expert [SAM63]. Book moves of recorded games played by experts were substituted for using an actual person. This provided Samuel with several hundred thousand training instances. A preference was given to games that led to both sides having about equal advantage at the end of the game. The book-move training was used to learn a set of signature tables. These were used to assign credit and blame to measured board features, in accordance with the extent the measured board features correctly predicted the expert's move.

Samuel's measured board features were abstract aspects of the board situation, such as total mobility and back-row control. These may approximate the type of features used by human experts in evaluating the merits of alternative moves for a particular board position. Analysis of protocols from checker experts might have enabled Samuel to determine how the features selected for each of the seven chronological phases of the game correspond to those that checker experts believed were most relevant. Such an analysis might also have been useful in determining if the initial set of thirty-two features were the most relevant features for evaluating alternative moves. However, even then, performance improvements might be limited because of the basic feature approach taken. Checkers experts probably chunk their knowledge around board configurations similar to the way chess experts are believed to organize knowledge. This organization provides a context for storing the long range strengths and weaknesses of particular positions.

Waterman's poker player infers a model of good play in its implicit training mode [WAT70]. Based on a retrospective analytic evaluation of a hand, the program determines the action it should taken at each point in the game. This provides a move-by-move performance standard. Each time there is a difference between the move the program would have made and the performance standard, a training rule is built by the learning element to modify the move that the program will make in future similar circumstances. The learned behavior is represented as production rules.

There is a way in which Davis's Teiresias program can be viewed as part of a learning by watching system [DAV76]. However, in this example, it is the expert that is inferring a model of the program and then comparing the model to the experts own model, rather than the program constructing a model of the expert. This relieves the program of having to face the difficult apportionment of credit problem: the expert provides this information. In many ways, our program is the exact reverse of Teiresias [BAR77].

7 Concluding Remarks

Our research on inferring a user's diagnostic reasoning is based on principled lines. We emphasize the creation of a learning environment similar to that in which humans are immersed during acquisition of expertise. Expertise is acquired starting with a rich base of textbook type knowledge, and then augmenting this knowledge through an apprenticeship period by watching seasoned expert's and trying one's hand as the expert problem solving. The environment being created approximates these conditions. Our approach also stresses the importance of using knowledge organization and inference techniques similar to those used by humans. The large body of literature on how physicians organize and use their knowledge during medical diagnosis provides us with the opportunity to construct an expert system that simulates this clinical reasoning process. This approach should lead to expert systems that escape the brittleness of current expert systems.

8 Acknowledgements

The organization of this paper was significantly altered by comments on earlier drafts provided by Roy Rada, M.D., and Marianne Winslett Wilkins. Thanks to Diane Hasling and Mark Richer for discussions relating to this paper. We express gratitude to our domain expert, Curt Kapsner, M.D.

This work was supported in part by NSF grant MCS-83-12148 and by ONR/ARI contract N00014-79C-0302. Computational resources were provided by by the SUMEX-AIM facility (NIH grant RR 00785).

9 References

[BAR77] Barr, A., Bennett, J. and Clancey, W., Transfer of expertise: A theme for AI research, Working Paper HPP-79-11, Stanford University, 1979, 6 pp.

[BUC83] Buchanan, B. G., Mechanizing the search for explanatory hypotheses, in *Philosophy of Science* **52** (1983) 129-146.

[CLA81] Clancey, W. J. and Letsinger, R., "Neomycin: Reconfiguring a rule-based system for application to teaching, *IJCAI-81*, 1981, 829-836.

[CLA83] Clancey, W. J., The epistemology of a rule-based system: A framework for explanation, *Artificial Intelligence* **20** (1983) 215-251.

[CLA84] Clancey, W.J., Acquiring, representing and evaluating a competence model of diagnosis, HPP Memo 84-2, Stanford University, 1984.

[CLA84b] Clancey, W. J., Classification problem

solving, *AAAI-84*, 1984, (in press).

[DAV76] Davis, R. and Lenat, D., *Knowledge-based systems in artificial intelligence*, New York: McGraw-Hill, 1982.

[ELS78] Elstein, A. S., Shulman, L. S. and Sprafka, S. A., *Medical Problem Solving: An analysis of clinical reasoning*, Cambridge: Harvard University Press, 1978.

[HOL75] Holland, J. H., *Adaptation in Natural and Artificial Systems*, Ann Arbor: University of Michigan Press, 1975, 183 pp.

[KAS82] Kassirer P., Kuipers, B. J. and Gorry, G. A., Toward a theory of clinical expertise, *Amer. Jour. Medicine*, **73** (August 1982) 251-259.

[LAN83] Langlotz, C. P. and Shortliffe, E. H., Adapting a consultation system to critique user plans, Memo HPP-83-2, April 1983, 19 pp.

[LIN80] Lindsay, R. K., Buchanan, B. G., Feigenbaum, E. A., and Lederberg, J., *Applications of artificial intelligence to organic chemistry: The Dendral project*, McGraw-Hill, New York, 1980.

[MIL75] Miller, P. B., Strategy selection in medical diagnosis, TR MAC-TR-153, MIT, 1975, 130 pp.

[MIT83] Mitchell, T., Utgoff, P. E. and Banerji, R. S., Learning by experimentation: Acquiring and refining problem solving heuristics, in Michalski, R. J., Carbonell, J. G. and Mitchell, T. M. (eds.), *Machine Learning*, Tioga Press, 1983, 572 pp.

[PAT81] Patil, R. S., Szolovits, P. and Schwartz, W. B., Causal understanding of patient illness in medical diagnosis, *IJCAI-81*, 1981, 893-899.

[PAU76] Pauker, S., Gorry, G., Kassirer, J. and Schwartz W., Toward a simulation of clinical cognition: taking the present illness by computer, *Amer. J. Medicine* **60** (1976) 981-995.

[POL84] Politakis, P. and Weiss, S. M., Using empirical analysis to refine expert system knowledge bases, *Artificial Intelligence*, **22** (1984) 23-48.

[POP82] Pople, H. E., Heuristic methods for imposing structure on ill-structured problems: The structuring of medical diagnostics, in Szolovits, P., (ed.), *Artificial Intelligence in Medicine*, Boulder: Westview Press, 1982, 119-190.

[SAM63] Samuel, A. L., Some studies in machine learning using game of checkers, in Feigenbaum, E. and Feldman, D. (eds), *Computers and Thought*, New York: McGraw-Hill, 1963.

[WAT70] Waterman, D., Generalization Learning Techniques for automating the learning of heuristics, *Artificial Intelligence*, **1** (1970) 121-170.