On Inference Process

Th. Westerhoff, Paderborn and P. Naeve, Bielefeld

1. Motivation

"An Expert system is a computing system capable of representing and reasoning about some knowledge-rich domain ... with a view to solving problems and giving advice." Jackson, p 1, (1986). Building such a system seems to be a manageable task considering the many expert system shells which are now available.

But anyone who is about to start building an expert system is well advised to take a word of warning, otherwise he migth find himself caught in a trap of undue problem reduction. Surely nobody willingly would accept the following definition of *intelligent*: "able to perform computer functions" Webster (1983). But the user of expert system shells could end up implicitly doing just this. The reasons for this are twofold. There are almost no papers describing the "inference process" of an expert in a way one could take as a design for building the inference component of a shell. On the other hand, expert system shells do not make their inference mechanism as explicit as one would need when modelling actual expert behaviour.

If one defines

Expert = Expert modeled in shell-elements PLUS remainder

it will turn out far too often that contrary to ones assumptions the remainder is neither small nor in any way controllable. So either for selecting a shell or for creating a new one a list of indispensable requirements is needed which when met by the shell allow for successful modelling of an expert's inference process.

This task was undertaken for the special case of a statistical expert. As an initial description language the terminology of the EMYCIN-paradigm was used to describe the expert's inference process.

The notation of "expert" used in this paper is embodid in several different sources of experience: the reflections and intuitions of a statistician (one of the author), observation by a

Compstat 1988
© Physica-Verlag Heidelberg for IASC
(International Association for Statistical
Computing) 1988

knowledge engineer (the other author), exploitation of similar attempts reported in the literature (especially the work of D.J. Hand) and last, but not least, the huge stock of statistical literatur revealing statistical expertise.

2. Requirements on an expert system shell

The result of the analysis of a "statistical" inference process can be divided into three main topics which are the headings of the following paragraphs.

2.1 Requirements of formalisms for representing knowledge and their use

In order to structure his knowledge, the statistician needs objects and rules. But what is characteristic of statistics is their dynamic use. Whereas the definition of objects and rules is always static, a statistician changes the ranges of values or even the values of a feature of an object in the course of consultation. Furthermore, he somtimes uses a rule for induction (a forward rule) and sometimes for deduction (a backward rule).

The dynamically changeable view of a statically defined (meta-) object is probably most specific to statistics. For example, a statistical expert changes his view of a statistical variable i.e. he adds or deletes attributes dependent on his current viewpoint. We suggest therefore, the following shell specifications for statistical expert systems:

1. The shell should admit objects and rules as knowledge representation formalisms.
2. It should be possible to change characteristics of objects (values, ranges of values) dynamically.
3. An object should allow for a dynamically changeable view of a static meta-object.
4. It should be possible to chain rules forwardly and/or backwardly.

2.2 Requirements for the inference engine

The different knowledge representation formalisms are often not crucial for the usability of an expert system shell; more crucial are the type and complexity of problem solving methods which can be implemented by it. The flexible use of inference processes is one of the main problems for expert system shells. Shells which were derived from existing expert systems by eliminating domain specific knowledge, inherit their problem solving mechanism from the domain-specific problem solving process. Therefore shells are often only usable for a reduced class of problems.

In the area of statistics one has to achieve a dynamic and flexible behaviour via the shell; there is no single predefined problem solving process for a statistical consultation. So we establish the following requirements:

5. The inference process should be controllable in such a way that any strategy can be modelled.
6. Deduced facts should be subject to deletion and revision, i.e. monotonic and non-monotonic reasoning and temporary inconsistencies should be possible.
7. The user, too, should be dynamically able to influence the process of inference.

2.3 Requirement for a facility for hypothetical reasoning

One of the most important requirements in representing statistical inference is the possibility of pursuing more than one goal at a time and drawing arguments by analogy. The following conclusions are drawn:

8. There should be possibilities to implement different worlds in a system.
9. Derived results of a world should be accessable in this world itself and in all its offspring worlds. But they may not influence parallel worlds.
10. It should be possible to jump from one world to a parallel one (sideward chaining). Therefore results of one world should be transferable to a parallel world.

3. Satisfaction of the requirements using two TWAICE-generations

In order to examine how existing expert system shells satisfy the given requirements, we looked into two "generations" of the Nixdorf expert system shell TWAICE, namely TWAICE release 2.5 and

TWAICE release 3.0. In the following, we only present features of TWAICE related to the above topics. Detailed descriptions of TWAICE can be found in Mescheder (1988) for TWAICE 2.5 and Schmitt (1988) for TWAICE 3.0. All statements about TWAICE not qualified by a release number are valid for both releases.

3.1 Knowledge representation formalisms and their use in TWAICE

In TWAICE, the taxonomy is used to structure the knowledge base by frame-based definitions of objects and attributes. Objects describe general types of topics (e.g. statistical variabale, statistical method). During a consultation certain individual objects are investigated which are called instances (e.g. the statistical variable "year"). Attributes describe features of objects and values describe their characteristics.

Rules formulate logical dependencies between the attributes of a domain. They are formulated for objects and can therefore be used repeatedly for different instances. There are backward and forward rules in TWAICE.

The knowledge sources of TWAICE, besides the obligatory rules and taxonomy, are optional tables, terms, procedures, and several user texts. In TWAICE 3.0 there are also optional facts, defaults, and databases.

As knowledge is mainly represented in objects and rules, the first requirement is fulfulled by TWAICE to the extent that inheritance of attributes between objects is not possible.

In TWAICE 2.5 all definitions of the taxonomy and rule base are static. Therefore this generation of TWAICE did not satisfy the dynamic requirements.

This is different in TWAICE 3.0, however. Independent of the static definition in the rule base, a rule can be chained forward and/or backward. Furthermore one can change the range of value of an attribute dynamically. So two of the three requirements of "dynamism" are fulfilled by TWAICE 3.0.

3.2 The inference strategy of TWAICE

The inference engine of TWAICE controlls the consultation and deduces results.

Its mode of work was iron-clad in TWAICE 2.5 and primarily used backward chaining. The predifined inference method uses monotonic logic. Furthermore it uses the principle of exhaustive evidence gathering. This means that in order to determine values of an attribute all relevant knowledge sources are taken into account.

A predifined inference strategy prevents one from influencing the course of problem solving dynamically. A particular strategy can only be reached by clever "programming by rules". This often requires a lot of awkward work for the knowledge engineer which prevents a "natural" modelling of the statistical expert.

TWAICE 2.5 was therefore unable to satisfy our requirements except in a primitive manner.
One of the most inportant differences between TWAICE 2.5 and TWAICE 3.0 is the opening of the inference component in TWAICE 3.0. Because of this opening a knowledge engineer can influence the inference process directly and modify all results of the dynamic knowledge base. This means that derived results or instances of objects may be deleted. Therefore non-monotonic reasoning and temporal inconstistencies are allowed in TWAICE 3.0. In order to influence the inference process directly a knowledge engineer is able to define his own special commands.

3.3 Hypothetical reasoning in TWAICE

In TWAICE 2.5 there is no concept analogous to a "world". Therefore hypothetical reasoning in TWAICE 2.5 is not possible.

In TWAICE 3.0 there exists the concept of "situations". Through situations one can partition the dynamic database into different worlds. TWAICE 3.0 manages a graph of these situations. Therefore situations in different branches of the graph can be independent while there are connections in the branches themselves. In particular, all results of a situation are inherited by its offspring unless explicitly deleted or changed.

The concept of "situations" in TWAICE 3.0 satisfies the requirements for hypothetical reasoning.

4. Summary

The representation of "statistical" inference requires dynamic behaviour by the main components of an expert system shell: the data structures for knowledge representation and the inference component. Furthermore the possibility for hypothetical reasoning and reasoning by analogy is mandatory.

Like probably every other expert system shell with predefined problem solving method, release 2.5 of TWAICE seemed not to be suitable for simulating statistical inference. The evolution of this shell to TWAICE 3.0 shows that by opening the inference component of the shell the direct representation of statistical inference becomes possible.

References:

Hand, D.J., Statistical expert system: necessary attributes, Journal of applied Statistics, vol 12, 1984

Hand, D.J., Statistical expert system: design, The Statistician, vol 33, 1984

Hand, D.J., Choice of statistical technique, ISI Bulletin of the International Statistical Institute, Proceedings of the 45th Session, vol LI/book 3, Amsterdam, 1985

Hand, D.J., Patterns in statistical strategy: in Artificial Intelligence & Statistics, Gale W.A. (ed), Reading, 1986

Jackson, P., Introduction to Expert Systems, Addison-Wesley, Wokingham, 1986

Mescheder, B., Functionality and operational features of the expert system shell TWAICE in: Savory, S. (ed): Artificial Intelligence and Expert Systems, Chichester, Ellis Horwood Ltd., 1988, 60-91

Schmitt, F., Knowledge Engineering mit einer Expertensystem-Shell, NCAG, 1987, to appear in KIFS, 1987

Webster's Ninth New Collegiate Dictionary, Merriam Webster, Springfield, 1983