

Introducing expert critiquing systems

STURE HÄGGLUND

Department of Computer and Information Science, Linköping University, S-581 83 Linköping, Sweden (Email: StuHA@ida.liu.se)

Expert critiquing systems present an interesting and important complement to conventional expert systems. The basic idea is to emphasize the support for the user's own decision making rather than to let the system independently suggest a solution to a given problem. Thus it is assumed that the user initially proposes a decision or course of action. The system then reviews this suggestion relative to known circumstance, tries to evaluate the proposed solution, provides suggestions for improvements, draws attention to possible risks, indicates alternatives, and evaluates their merits, etc. A critiquing system is characterized both by its approach to problem solving and by its style of interaction with the user, as will be detailed below.

The work by Perry Miller (1986) in the area of medical applications, as summarized in his book *Expert Critiquing Systems* introduced the concept of expert critiquing to a wider audience. Related issues were also independently investigated by others, for instance in the context of the MYCIN tradition, where the need for resolving conflicts between an advice-giving program and the preferences expressed by a reasonably competent user were recognized (Clancey, 1977; Longlotz & Shortliffe, 1983). An interesting variant, albeit not conceived of explicitly as critiquing, was employed in the HELP system, where medical treatment decisions were supervised and alerts generated whenever contraindications were detected (Pryor, 1983). The feasibility of automating critique generation relying solely on medical records has been demonstrated in a series of studies showing that existing documentation of patient data was sufficient for generating useful critique (Van der Lei, 1991).

Early studies of similar ideas were also performed in the area of engineering design. This is an example of a domain where it is very difficult to automate or even mechanize decision making, due to the complexity of a typical design space. An example of early work in this domain is the CRITTER system (Kelly, 1984), which evaluates and comments upon digital circuit designs. Important contributions in the area of design critiquing have also been made by Fischer and colleagues, starting with the Lisp critic (Fischer, 1987) and leading up to general work on human-computer collaborative design support, as exemplified also by their paper in this special issue (Fischer, 1993).

A theory of expert errors and critiquing forms the basis for work by Silverman, resulting in the COPE system which can be viewed as a generic critic shell (Silverman, 1987). An extensive presentation of methodological and practical issues in critic engineering is given in Silverman (1992b). Aspects of usability in connection with expert critiquing have also been discussed in, for instance, Hägglund and Rankin (1988).

Numerous applications are developed today using the expert critiquing approach. For a recent overview of the area, see Silverman (1992a). The first international workshop dedicated to the topic of expert critiquing systems was held in conjunction with the AAAI-93 conference in Washington D.C. in July 1993. This workshop demonstrated the broad range of applicability for the expert critiquing approach, but also a large number of topics for further scientific studies in the area.

Several different types of motivations can be identified for the emerging interest in the expert critiquing approach to giving knowledge-based consultative advice, mediated by a computer system. The papers in this special issue illustrate and illuminate some major directions of research in the area.

The paper by Fischer et al. (1993) is focusing on the collaborative support for problem solving in areas where it is not interesting or realistic to aim for a complete expert system capable of solving problems in the given domain on its own. In this case, critiquing provide a means for creating systems in ill-defined domains with partial and evolving problem specifications, such as, for instance, typical design applications. The paper describes how the advisory system can be embedded in a domain-oriented design environment, providing *generic*, *specific* and *interpretative critics* which complement each other as mechanisms for obtaining a truly creative collaborative design support environment for the users. Their approach is illustrated with a presentation of the HYDRA architecture for design environments.

Silverman's paper (Silverman, 1993) illuminates another important aspect of the expert critiquing approach, namely the issue of handling expert bias and human error in decision making situations. Thus, Silverman presents a methodology for *critic engineering* that is useful in supporting knowledge acquisition and representation, with a special emphasis on decreasing the degree of biased judgement in the elicited knowledge. The critic-based approach is useful both during the knowledge engineering process and when using the resulting system for giving consultative advice to a user. The paper describes the methodology, and reviews a number of applications to which it has been applied.

Rankin, in his paper on text generation for expert critiquing (Rankin, 1993), covers still another area, namely the formulation of the argumentative support needed for the negotiation that typically takes place between the system and the user in a critiquing context. His paper reviews techniques for text generation, although most existing expert critiquing systems are still using simple canned-text techniques. Rankin, however, demonstrates that the real challenge is the deep generation of a critique, i.e., deciding upon what information to include, which argumentative structure to apply, and how to adapt the presentation to a particular user. He shows how the dominant tradition in current text generation research, Rhetorical Structure Theory, can be applied, but also which limitations it has. The paper explains how argumentation theory can be used as a framework for critiquing and justificatory texts, and also extends the discussion to dialogue situations by introducing a language game metaphor for critiquing.

In general, expert critiquing exhibits certain characteristic aspects, which are interesting to reflect upon. Thus such a system:

- can be based upon heuristic domain expertise or upon formal decision theories. This is the same kind of distinction that applies to expert systems in general. Most practical experience indicates that the heuristic approach provides a more convenient foundation for development of working systems, and the critiquing paradigm extends the applicability of heuristic methods for this purpose;
- can be primarily *knowledge-based* or *algorithmic*. It should thus be interesting to distinguish between knowledge-based approaches to critiquing, or *expert critiquing*, and algorithmic critics such as spell checkers;
- might employ either an *analytical* or a *differential* approach to critiquing. Taking the analytical approach means that aspects of the user's proposal for a decision are evaluated and commented upon, while the differential approach is based on a comparison of the proposed solution against some "correct" or standard solution. Depending upon the application, one or the other of these solutions are preferred or more easily adopted;
- may focus upon the generation of a *critique* of the user's preferred solution, but also upon a *justification* of the system's own proposals. In general, we ought to think of expert critiquing systems rather as systems capable of engaging in an argumentative discussion with the user;

- may use *ad hoc* or *linguistically sound* approaches to text generation. Especially important is the deep generation of a critique, and the rhetorical structures used for the discourse.

When considering the deliberations we have to undertake in the process of making a decision of whether to employ a critiquing or a conventional approach in a knowledge engineering project, the following observations may be helpful:

- It is typically a more complicated task to build an expert critiquing system than a conventional one for a specific application, since not only should we equip the system with enough knowledge to derive a preferred solution, but also, we need to give it the power to critique a broad variety of alternative approaches to handling a given problem.
- The scope of applications we can handle may be broader with a critiquing approach, since we can build useful systems even for applications where it is not possible to give comments to various proposals suggested by the user. That is, the system may not be proficient enough to generate complete solutions on its own.
- In cases where the system is capable of producing solutions that are expected to provide the "right" answer to a given problem, a critiquing approach may still be preferred for the purpose of better prospects for user acceptance.
- A critiquing system can conveniently handle conflicting expertise in a domain. Since the system doesn't have to come up with one specific recommended solution, it can maintain a knowledge base with alternative approaches for handling a given problem. If the analytical approach to critiquing is employed, a proposed solution can be easily evaluated with respect to these different approaches.
- A critiquing system is also useful with partial knowledge in the given domain of application. When only certain aspects of a problem can be handled by the system, or when the knowledge of a given situation lacks depth, the system can inform the user about the quality and reliability of the comments and critique generated.
- The expert critiquing approach offers transition support for developing a proficient expert system from a critiquing system with a limited knowledge base. Thus a conventional expert system cannot be used in practice before the knowledge base is reasonably complete and validated. Here the critiquing approach offers a possibility to get practical experience from using the system already with a partial knowledge base. Once the knowledge base is complete enough, it can be used as a problem-solving aid in the style of traditional expert systems, at least given that a differential approach to critiquing was supported.

In our view, one might also distinguish between the following different use of critics and critiquing:

- Critics used as a mechanism for reasoning and problem solving. In this case, the main use of critiquing is as a way of organizing the knowledge and inferencing needed to come up with a proposal for handling a given problem. The knowledge base is typically organized as a set of critics that apply in different situations, and these critics may, for instance, be used in a generate-and-test approach to problem solving. An example of a possible application is for critiquing software specifications (Fickas, 1988).
- Critiquing used as a way of providing non-intrusive advice to a computer user. In this case, the main purpose of the critiquing approach is to ensure that the user is in control of the decision making, and the system restricts itself to commenting upon the user's suggestions, even if it were possible for the system to come up with a preferred solution of its own. The main focus is on the process of interaction between the system and user, as exemplified, for instance, by Fischer (1993).
- Critiquing viewed as the basis for presenting arguments and explanations in an understandable and effective way to a given user. In this case, the main focus is on the contents of a critique, its extension, components and rhetorical structure. Work on text generation relates to this aspect of critiquing (Rankin, 1993).

The current developments in the area of applying the expert critiquing paradigm for building truly collaborative, user-friendly knowledge-based systems presents an interesting and exciting challenge. It is our hope that the papers in this special issue of *The Knowledge Engineering Review* should contribute to this development.

References

- Clancey, WJ, 1977. "An antibiotic therapy advisor which provides for explanations" In: *Proceedings of the 5th International Joint Conference on Artificial Intelligence* p 858, Cambridge, MA.
- Fickas, S and Nagrajan, P, 1988. "Critiquing software specifications" *IEEE Software* 5 (6) 37–47.
- Fischer, G, 1987. "A Critic for LISP" In: *Proceedings of the 10th International Joint Conference on Artificial Intelligence* pp 177–184, Milan, Italy.
- Fischer, G, Nakakoji, K, Ostwald, J, Stahl, G and Summer, T, 1993. "Embedding critics in integrated design environments" *The Knowl. Eng. Rev.* 8 (4).
- Hägglund, S and Rankin, I, 1988. "Investigating the usability of expert critiquing in knowledge-based consultation systems" In: *Proceedings of the ECAI'88 Conference* pp 152–154, Munich, Germany.
- Kelly, VE, 1984. "The CRITTER system: Automated critiquing of digital circuits design" In: *Proceedings of the 21st Design Automation Conference* pp 419–425.
- Langlotz, CP and Shortliffe, EH, 1983. "Adapting a consultation system to critique user plans" *Int. J. Man-Machine Studies* 19 479–496.
- Miller, P, 1986. *Expert Critiquing Systems*, Springer-Verlag.
- Pryor, TA, Gardner, RM, Clayton, PD and Warner, HR, 1983. "The HELP system" *J. Medical Syst.* 7 (2) 87–102.
- Rankin, I, 1983. "Natural language generation in critiquing" *The Knowl. Eng. Rev.* 8 (4).
- Silverman, BG et al., 1987. "COPE: A Case Oriented Processing Environment" In: *Proceedings of the European Computing Conference*, Avignon, France.
- Silverman, BG, 1992a. "Survey of expert critiquing systems: Practical and theoretical frontiers" *Comm. ACM* 35 (4) 106–127.
- Silverman, BG, 1992b. *Critiquing Human Error. A Knowledge Based Human-Computer Approach*, Academic Press.
- Silverman, BG, 1993. "Engineering expert critics for cooperative systems" *The Knowl. Eng. Rev.* 8 (4).
- Van der Lei, J, 1991. *Critiquing based on computer-stored medical records*, PhD dissertation, Erasmus University, Rotterdam.