PROBLEM SOLVING AND SUSTAINED LEARNING FROM EXPERIENCE: ANALYZING METHODS WITH RESPECT TO DOMAIN CHARACTERISTICS

Agnar Aamodt
University of Trondheim
Dept. of Informatics
N-7055 Dragvoll, Norway
email: agnar@ifi.unit.no

Klaus-Dieter Althoff
University of Kaiserslautern
Dept. of Computer Science
D-67653 Kaiserslautern, Germany
email: althoff@informatik.uni-kl.de

Introduction

Integration of learning and problem solving may start out from different goals, and be viewed from different perspectives. One example is "concept formation" as a goal, and the formation and utilization of operationalization criteria related to the problem solving solving task, as the perspective. Another example is "improved performance" as a goal, and the improvement of total problem solving speed - for computer and human together - as the perspective. A third example is "sustained learning", i.e. continuous learning through problem solving experience, as a goal, and the impact of the application problem task on the learning method as a perspective. Many more examples may be given, and for each of them a particular area of overlap, an "intersection space" between machine learning (ML) and problem solving (PS) methods can be identified. Within this space, interdependencies between ML and PS methods may be described and analyzed in a systematic way, provided we have a suitable means to structure the space.

We have initiated work towards the development of a framework which defines and makes use of such a structure. The focus is on the third of the above goals: Sustained and (continuous) learning from each problem solving experience as a natural part of a problem solver's behavior. Moreover, we want to start out by studying the simplest and most direct form of experiential learning: Retaining of experiences as concrete, non-generalized problem solving episodes, and corresponding problem solving by the method of case-based reasoning (CBR). This is motivated, firstly, by that it is natural to start a study of learning as close as possible to the source of the learning, namely the concrete experience. Further it is motivated by the experimental results of the ML field so far, which have shown that the automatic learning of generalizations in all but rather simple and well-defined domains is extremely difficult (Porter et al., 1990; Utgoff, 1986; Lebowitz, 1990; Bergadano et al., 1991).[1]

The aim of our research is twofold. The first is related to AI as an experimental science: To establish an analytic framework for improved understanding of the PS-ML integration problem, discussion of possible solutions, analysis of experiments and comparison of results. The second is related to knowledge-based systems engineering: To specify ML-PS integration as an important task to be handled by the knowledge engineer, and to provide a method for dealing with this problem. In this contribution, we summarize the basic ideas of the framework as developed so far, and start a discussion of how it can be used to analyze and design problem solvers that sustainly learn by experience.

A Framework for Integrated Problem Solving and Sustained Learning

A potentially useful way to describe problem solving behavior is in terms of the goals that a problem solver has, the tasks that are generated to achieve the goals, the methods that will accomplish the tasks, and the knowledge of the application domain that these methods need. A description of a system along these lines is often referred to as a knowledge level description, and recent research in knowledge acquisition (e.g. Steels, 1992; Wielinga et al., 1993) has clearly demonstrated the usability of this approach. The original knowledge-level idea has undergone some modifications over the years, from Newell's highly intentional, purpose-oriented way of describing a system (Newell, 1982), to a somewhat more structured and usable type of description. This transition has also led to modifications of the knowledge notion itself, associated with terms such as the "knowledge use level" (Steels, 1990), a "knowledge level architecture" (Sticklen, 1989), and the notion of "tractable rationality" (Van de Velde, 1988). The original notion of knowledge level has been extended by introducing high-level structural and methodological constraints. This makes the knowledge level more operational and useful for conceptual modeling purposes, while retaining its competence-oriented and implementation-independent aspects. In this contribution, the framework basically operationalizes a knowledge-level description of the integrated problem solving and learning task for case-based reasoning.

The main purpose of adopting a knowledge-level perspective to the analysis of integrated PS-ML systems is to use it as a means of scientific communication. Primitive methods can achieve an intuitive kind of "semantics" by refering to the realization of known algorithms within already described/implemented systems (e.g., case-based reasoning is combined with rule- and model-based reasoning in the same way as in CREEK (Aamodt, 1991); a decision tree is generated as in INRECA (Manago et al., 1993); the similarity measure is adapted as in PATDEX (Wess, 1993); partial determination rules are generated and used like the so called "shortcut rules" in MOLTKE (Althoff, 1993); etc.)[2]. In addition, a mid-term purpose is, after having developed/described a certain number of systems, to be able to select/instantiate a system by starting at the knowledge level. Then a symbol-level architecture can be chosen, and/or a chosen architecture can be instantiated, based on a better understanding of the real world task and its domain characteristics. A knowledge-level description in itself, of course, does not provide a language and suitable means to arrive at a symbol-level architecture specification. In our framework, we therefore introduce a focusing perspective and an analytic "tool" to help in the more detailed description that guides the architectural specification based on a knowledge-level model.

The focusing mechanism is to view in principle all CBR methods as operations related to similarity, in one sense or another. That is, problem solving can be described as a process of initial assessment of similarity (case retrieval) followed by a more de-

---

[1] Nevertheless we will also include generalization within our framework at a later stage, e.g. by focusing on "lazy generalization" like in PROTOS (Bareiss et al., 1987) or GenRule (Althoff et al., 1989).

[2] The level of detail which is used in this references is mainly depending on the communication purpose and on what is already described and/or known in the community. In principle, it could be as detailed as necessary.

liberate assessment of similarity (case adaptation), and learning (case extraction, case indexing, also updates of general background knowledge) can be described by relating it to later similarity assessment - i.e. to a pragmatic learning goal. Along with Richter and Wess (1991), similarity is viewed as an a posteriori criterion, and attempts to assess similarity before a retrieved case has been found useful, will always give a hypothesized similarity assessment. A way to describe the role of general domain knowledge (background knowledge), for example, is then as a means to reduce the uncertainty of the initial similarity assessment (leading to the retrieval of a case) with respect to the final similarity assessment made after having evaluated the success of the (possibly modified) case in finding a solution to the current problem.

The domain types addressed cover a wide spectrum, ranging from strong-theory domains with rather well-defined domain relationships (e.g. diagnosis of purely technical systems), to weak-theory, open domains (e.g. medical diagnosis). This is an important feature of the framework, since we particularly want to relate characteristics of the task (the what-to-do) and the knowledge on which performance of the task is based, to the methods (how-to-dos) that enable the problem solver to accomplish the task by use of the knowledge. A reason to focus on this mapping, is that for many real world problems, the subtasks and domains involved are of different types, and we need to integrate reasoning based on strong and weak knowledge, as well as different application task types. Medical diagnosis, for example, would have to be moved far away from its real world context, for it to be regarded as a pure classification task. If a system shall cover the major tasks involved in practical diagnosis, it will have to include planning tasks (e.g. setting up and continuously revising an examination protocol), as well as prediction tasks (assessing the consequences of a treatment).

The top level view is as follows: The overall goal of the system we want to describe, then is to continually improve the quality of problem solving. The method for solving the top-level task is case-based reasoning, which splits the top-level task into two subtasks: Problem Solving and Sustained Learning, with the corresponding methods Solve-problem and Learn-from-the-experience.

References

Aamodt, A. (1991). A knowledge-intensive approach to problem solving and sustained learning. Ph.D. Dissertation, University of Trondheim, Norwegian Institute of Technology, May 1991

Aha, D., Kibler, D. & Albert, M. K. (1991). Instance-based learning algorithms. Machine Learning, Vol. 6, No. 1, Jan. 1991

Althoff, K.-D., Faupel, B., Kockskämper, S., Traphöner, R. & Wernicke, W. (1989). Knowledge Acquisition in the Domain of CNC Machining Centers: the MOLTKE Approach. In: Boose, Gaines & Ganascia (1989), *Proc. EKAW-89*, pp 180-195

Althoff, K.-D. (1993). *A Case-Based Learning Component as an Integrated Subpart of the MOLTKE Workbench for Fault Diagnosis in Engineering Systems (in German: Eine fallbasierte Lernkomponente als integrierter Bestandteil der MOLTKE-Werkbank zur Diagnose technischer Systeme)*. St. Augustin (Germany): DISKI 23, infix Verlag

Bareiss, R. & Porter, B. (1987). PROTOS - an exemplar-based learning apprentice. Proc. 4th International Workshop on ML, Irvine, June 1987, pp 12-23

Bergadano, F., Giordana, A. & Saitta, L. (1991). *Machine Learning - an integrated framework and its applications*. Ellis Horwood Limited

Hammond, K. (1986). CHEF - a model of case-based planning. Proc. of AAAI-86, Morgan Kaufmann, pp 267-271

Koton, P. (1989). Using experience in learning and problem solving. MIT, Laboratory of Computer Science Ph.D. Dissertation, October 1988), MIT/LCS/TR-441

Lebowitz, M. (1990). The utility of similarity-based learning in a world needing explanation. In: Y. Kodratoff & R. S. Michalski (eds), Machine Learning- an AI approach, Vol. III, Morgan Kaufmann, pp 399-432

Manago, M., Althoff, K.-D., Conruyt, N., Maurer, F., Traphöner, R. & Wess, S. (1993). Induction and Reasoning from Cases. To appear in: *Proc. EWCBR-93*

Newell, A. (1982). The knowledge level. *Artificial Intelligence*, Vol. 18, pp 87-127

Plaza, E., Aamodt, A., Ram, A., Van de Velde, W. & van Someren, M. (1993). Integrated Learning Architectures. *Proc. ECML-93,* pp. 429-441

Plaza, E. & Arcos, J.-L. (1993). Reflection, memory, and learning. Institut d'Investigacio en Intelligencia Artificial, CSIC/CEAB, Report de Recerca IIIA 93/2. Also in Proc. MSL-93 Workshop on Multistrategy Learning

Porter, B., Bareiss, R. & Holte, R. (1990). Concept learning and heuristic classification in weak theory domains. *Artificial Intelligence, Vol. 45, No. 1-2*, Sept. 1990, pp 229-263

Richter, M. M. & Wess, S. (1991). Similarity, uncertainty and case-based reasoning in Patdex. In: R. S. Boyer (ed.), Automated Reasoning - Essays in Honor of Woody Bledsoe, Kluwer Academic Publishers, pp. 249-266

Steels, L. (1990). Components of Expertise. *AI Magazine*, 11(2), Summer 1990, pp 29-49

Steels, L. (1992). Reusability and configuration of applications by non-programmers. *VUB AI Memo 92-4*, Free University of Brussels (VUB)

Utgoff, P. E. (1986). *Machine learning of inductive bias.* Kluwer Academic Publishers

Van de Velde, W. (1988). *Learning from experience*. Ph.D. thesis, Free University of Brussels (VUB)

Van de Velde, W. & Aamodt, A. (1992). Machine Learning Issues in CommonKADS. KADS-II Report, KADS-II/Tii.4.3/TR/VUB/002/3.0, Free University of Brussels (VUB)

van Someren, M., Zheng, L. L. & Post, W. (1990). Cases, models or compiled knowledge ? A coparative analysis and proposed integration. In: B. Wielinga et al. (eds.), Current Trends in Knowledge Acquisition, IOS Press, 1990, pp 339-355

Wess, S. (1993). PATDEX - Knowledge-based, Incremental Improvement of Similarity Measurement in Case-Based Diagnostic Problem Solving (in German: PATDEX - ein Ansatz zur wissensbasierten und inkrementellen Verbesserung von Ähnlichkeitsbewertungen in der fallbasierten Diagnostik). *Proc. of the 2nd German Conference on Expert Systems,* Hamburg

Wielinga, B., Van de Velde, W., Schreiber, G. & Akkermans, H. (1993). Towards a unification of knowledge modeling approaches. *Knowledge Acquisition Journal* (forthcoming)