Overview of Logic-based Learning in Germany

Katharina Morik
Univ. Dortmund, Computer Science VIII,
e-mail: morik@kimo.informatik.uni-dortmund.de

March 30, 1998

The capability of learning is a central feature of every intelligent behavior. If we demand intelligent behavior of computer systems we have to incorporate some learning ability into programs. Already at the beginning of computer science, Alan Turing proposed that computers should learn from a human teacher [Turing, 1959]. Nowadays, machine learning has become a key issue of artificial intelligence[1]. The special interest group for machine learning within the German society for computer science (FG 1.1.3 in GI e.V.) that we started in 1988 has about 600 members. The members are coming from computer science, cognititive sciences, and industries. This indicates the different interests in machine learning: the interest in more intelligent systems (the software engineering aspect), the interest in formal properties of learning and learnability (the theoretical aspect), the interest in more efficient and effective work (the applicational aspect), and the interest in human learning abilities (the interdisciplinary cognitive aspect).

There are several frameworks in which learning and learnability can be formalized. Logic is one of them. The inductive learning problem of logic-based learning, which we want to deal with[2], is:

Given: a theory T and a set of examples $E := E^+ \cup E^-$
where $\exists e_+ \in E^+, T \not\models e_+$,
$\forall e_- \in E^-, T \not\models e_-$

Goal: a hypothesis H that fulfills the following conditions:
$\forall e_+ \in E^+, T \cup H \models e_+$,
$\forall e_- \in E^-, T \cup H \not\models e_-$,
$T \cup E \not\models \neg H$, $T \cup H \not\models \Box$.

In this talk, the different aspects of machine learning will be elaborated with respect to logic-based learning, using our own work for illustration. First, a paradigm of integrating machine learning into knowledge acquisition is presented. Second, logic-oriented learning is sketched. Two learning algorithms, RDT and KLUSTER, show alternative ways to restrict predicate logic in order to make the learning problem feasible. Theoretical results on lower bounds of complexity regarding logic-based learning are indicated. Third, applications of machine learning are illustrated by real-world applications of the MOBAL system. Finally, current interdisciplinary research on conceptual models of the day/night cycle is introduced.

[1] The section with the most papers at the International Joint Conference on Artificial Intelligence 1991 was the one on machine learning. In addition, there are three international conferences on computational learning: the conference on machine learning (IML), the one on computational learning theory (CoLT), and the one on algorithmic learning theory (ALT), not to mention the workshops on particular topics of learning. European and national conferences accomplish the activities of the field.

[2] Other learning problems that are formalized in logic are the inversion of resolution [Wirth, 1989] and the incremental inductive learning in the limit [Angluin and Smith, 1983].

1 Integrating machine learning into knowledge acquisition

The first research project on machine learning in the Federal Republic of Germany, LERNER, was funded by the Ministry for Research and Technology (BMFT) started in 1985, and ended in 1989. It was a collaboration of the Technical University Berlin, the software house Stollmann GmbH, and the Nixdorf Computer AG. The aim of the project was to integrate machine learning into knowledge acquisition[3]. A new paradigm of knowledge acquisition was introduced, the framework of **sloppy modeling** [Morik, 1989]. We wanted a new type of system that assists a knowledge engineer in building and maintaining a domain model. The user should be in control of how he or she wants to organize the modeling. The modeling process was recognized as an infinite process of enhancing a current model. The user should be allowed to be "sloppy", that is, make mistakes or input preliminary propositions and later on correct or enhance them with the help of the system. Each model is a "sloppy" model when compared to its successor. A system supporting the infinite modeling cycle needs a high-level knowledge representation that eases the structuration and inspection of the evolving model. Moreover, the user must be supported in updating and revising the current model. The BLIP system became the first implementation of a system that handles sloppy modeling [Morik, 1987]. It incorporated the METAXA learning algorithm and an inference engine [Emde et al., 1983]. This system had already moved beyond the attribute-value representation employing a restricted higher-order logic.

The integration of machine learning and knowledge acquisition can be realized as balanced cooperative modeling of system and user [Morik, 1993]. The learning component is one of several tools that assist the user in structuring, enriching, and revising a knowledge base. The tools are embedded in a knowledge representation environment. In the MOBAL system[4] knowledge is represented using rules and facts in a many-sorted logic. The rules are restricted to Horn clauses. MOBAL offers five modeling tools to the user. Each can support the user in performing a particular modeling task or perform that task itself.

- Structuring the arguments of predicates: the user may declare the sorts of predicate arguments, or the **sort taxonomy tool** may compute classes of sorts on the basis of given facts [Kietz, 1988]. In each case, the sort-correctness is maintained. A lattice of user-given sorts or classes of sorts is computed.
- Structuring the predicates with respect to the rules: the user may form groups of predicates and introduce directed links between those groups, where a correct rule must consist of predicates from the same group or have a predicate from a superior group in its conclusion. This topology of predicates gives an overview of the rule base. It can be automatically constructed by the **predicate structuring tool** [Klingspor, 1991].
- Verifying new rules: the user may input rules and have them checked with respect to corresponding facts by the system, or the **rule discovery tool** may learn new rules on the basis of given facts [Kietz and Wrobel, 1991].
- Revising rules because of contradictions: if the inference engine derives contradictory facts, the **knowledge revision tool** presents the derivation of the contradiction to the user so that s/he chooses what to delete and what to keep, or the tool performs a minimal change [Wrobel, 1993].
- Introducing new predicates and defining them: sometimes a new predicate is needed in order to separate rule applications that have led to a contradiction from successful applications of the same rule. The user may declare this new predicate and provide the corresponding facts, or the **concept learning tool** learns the definition of such a predicate [Wrobel, 1988].

[3] In 1992, a project for integrating case-based reasoning into a KADS type of knowledge acquisition was started by the BMFT: the FABEL project.

[4] MOBAL is the successor of BLIP. It has been developed at the German National Research Center for Computer Science in the course of the European project Machine Learning Toolbox (P2154). A detailed description of MOBAL is [Morik et al., 1993b].

Machine learning does not replace a knowledge engineer in modeling. Instead, inductive algorithms in concert with other techniques are capable of assisting the knowledge engineer.

2 Logic-oriented machine learning

Predicate logic allows to write statements that are easier to understand than their attribute-value compilation. Moreover, some statements cannot be expressed in propositional logic. Learning in predicate logic has therefore become a hot topic. However, learning formulas in full predicate logic is not feasible. The inductive learning problem in predicate logic inherits the semi-decidability and complexity from deductive reasoning in predicate logic. Therefore, it is an important research issue to determine restrictions of predicate logic that are as close as possible to predicate logic, but are tractable. The Rule Discovery Tool, RDT, of MOBAL uses function-free Horn clauses for representing the learning results and restricts the hypothesis space for learning by rule schemata. A rule schema expresses the syntactical form of rules. At least one predicate variable occurs in the place of a predicate symbol. The user specifies the overall set of learnable rules by inputting some rule schemata. Given a set of rule schemata the hypothesis space is then restricted to instantiations of predicate variables by predicate symbols of the same arity[5]. The rule schemata are partially ordered with respect to generality. RDT learns the most general hypotheses fulfilling the user-given acceptance criterion[6]. RDT learns in a top-down manner, testing instantiations of the more general rule schemata before either instantiating specializations of the schemata or pruning the search.

In Germany, several projects work on term-subsumption formalisms. The first algorithm that learns such concept definitions from facts is KLUSTER [Morik and Kietz, 1989]. It has been proved that KLUSTER learns in polynomial time [Kietz and Morik, 1993]. No further extension of the representation formalism is possible without losing this property. This is an upper bound for logic-based learning. Another one has been proved by Muggleton and Feng: it is possible to learn ij-determinate Horn-clauses in polynomial time [Muggleton and Feng, 1992], where i fixes the depth of chaining and j fixes the arity of predicates in the clause. Now, the first results concerning lower bounds have been achieved: learning determinate clauses with unbounded i is in the complexity class of PSPACE; and learning 12-indeterminate clauses is NP-hard [Kietz, 1993]. What particular assumptions decrease the complexity of learning or, in other words, make it easier to learn? One assumption is that the algorithm can ask questions. Asking equivalence questions to a teacher makes formulas with k literals learnable [Angluin, 1988]. Further theoretical work on the effect of realistic assumptions concerning the learning task is needed.

3 Applications of logic-oriented learning

In Germany, there is a company making its living by machine learning - Brainware GmbH at Berlin. In diverse applications they put different learning algorithms and combinations of algorithms to good use. For instance, for a very data-intensive application, Brainware used a combination of neural network learning and inductive learning where each algorithm alone was unable to achieve a result. The combination of the algorithms enabled to achieve results that were 20-25% better than those achieved by statistical techniques [BrainwareGmbH, 1991]. Other companies offer products[7] or services. Applying machine learning in real world domains has the potential to save resources in building-up and maintaining knowledge-based systems as well as technical systems, to increase throughput in factories, and to cut administrative efforts [Morik, 1992].

[5] This syntactic characterization of rule sets is similar to the extended Horn clauses of Yokomori [Yokomori, 1986]. A unifying framework for declarative bias is currently under development at the university Stuttgart by Birgit Tausend.

[6] Normally, the criterion is that the hypothesis should cover all positive and no negative examples, but the user may also tolerate a ratio of covered negative examples.

[7] The Krupp Technologie Transfer GmbH at Duisburg markets the learning program RuLEarn.

There are many ways to apply logic-based learning to real-world problems. In a robotics application, MOBAL has learned action-oriented object descriptions from sensor data [Morik and Rieger, 1993]. From experiencing some doors while following a path, the general concept of a door is learned. This will enable the robot to accept commands such as "Move through the door"in a new environment. This, in turn, eases the interaction with robots and makes robot applications more flexible.

Together with FORTH two medical domains, maldecensus testis and abdominal pain, have been modeled. MOBAL was used to correct case data on the basis of background knowledge and to correct and enhance the knowledge using case data [Morik et al., 1993a]. Together with Siemens AG, we have modeled a power supply network with respect to the messages that arrive at the network control center. The advantage of usig MOBAL as opposed to a classical expert system environment is the ease of changing the domain model. A case study on modeling the power system of a satellite was performed by British Aerospace. This company also used MOBAL as an aid in building up a knowledge base for design of aircraft parts [Parsons and Puzey, 1992]. The most challenging application so far is the modeling, updating, and maintaining of security policy for telecommunication networks that has been undertaken together with Alcatel Alsthom Recherche [Sommer et al., 1992]. In this application, the knowledge revision capability that uses a minimal specialization operator is more important than the rule discovery [Wrobel, 1993]. This again shows that use of machine learning should not be restricted to constructing rule sets. Further work on integrating machine learning techniques into standard software environments such as databases are an important task of future work. Also some help and guidance in customizing tools to a particular application are necessary in order to exploit the potential of the technique, namely that the system can be easily handled. Our first step into that direction was a human-computer interface that is customized to the security policy application.

4 Human and machine learning

Artificial intelligence in Germany always took into account the cognitive perspective. With respect to human and machine learning, a special Ph. D. program has been installed at the university of Freiburg at the beginning of this year. Another activity is coordinated by the European Science Foundation within its program "Machine and Human Learning". In collaboration with a psychologist who investigated explanations of the day-night cycle provided by children at different age [Vosniadou and Brewer, 1993], we represented the mental models within the MOBAL system. The formal representation clarifies the psychological models and allows for systematical experiments that are not feasible with human subjects. Currently, we are designing experiments that could clarify sequencing effects (e.g., which concept must be learned before another one can be obtained?). Determining appropriate sequences of instructions is relevant for teaching as well as for decreasing the computational complexity of machine learning.

References

[Angluin, 1988] Angluin, D. (1988). Queries and concept learning. *Machine Learning*, 2:319–342.

[Angluin and Smith, 1983] Angluin, D. and Smith, C. (1983). Inductive inference: Theory and methods. *Computing Surveys*, 15:237–269.

[BrainwareGmbH, 1991] BrainwareGmbH (1991). Machine induction on complex signals. *Pragmatica*, page 14.

[Emde et al., 1983] Emde, W., Habel, C. U., and Rollinger, C.-R. (1983). The discovery of the equator or concept driven learning. In *IJCAI-83*, pages 455 – 458, Los Altos, CA. Morgan Kaufman.

[Kietz, 1988] Kietz, J.-U. (1988). Incremental and reversible acquisition of taxonomies. *Proceedings of EKAW-88*, pages 24.1–24.11. Also as KIT-Report 66, Technical University Berlin.

[Kietz, 1993] Kietz, J.-U. (1993). Some lower bounds for the computational complexity of inductive logic programming. In Brazdil, P., editor, *Machine Learning - Proceedings of ECML-93*, volume 667 of *Lecture Notes in Artificial Intelligence*, pages 115 –123, Berlin, Heidelberg, New York. Springer.

[Kietz and Morik, 1993] Kietz, J.-U. and Morik, K. (1993). A polynomial approach to the constructive induction of structural knowledge. *Machine Learning*, 11.

[Kietz and Wrobel, 1991] Kietz, J.-U. and Wrobel, S. (1991). Controlling the complexity of learning in logic through syntactic and task-oriented models. In Muggleton, S., editor, *Inductive Logic Programming*, chapter 16, pages 335 – 360. Academic Press, London. Also available as Arbeitspapiere der GMD No. 503, 1991.

[Klingspor, 1991] Klingspor, V. (1991). Abstraktion von Inferenzstrukturen in MOBAL. Master's thesis, Univ. Bonn.

[Morik, 1987] Morik, K. (1987). Acquiring domain models. *Intern. Journal of Man Machine Studies*, 26:93–104. also appeared in Knowledge Acquisition Tools for Expert Systems, volume 2, J. Boose, B. Gaines, eds., Academic Press, 1988.

[Morik, 1992] Morik, K. (1992). Applications of machine learning. In Wetter, T., Althoff, K.-D., Boose, J., and Gaines, B., editors, *Current Developments in Knowledge Acquisitiobn - EKAW92*, volume 599 of *Lecture Notes in Artificial Intelligence*, pages 9 – 13.

[Morik, 1993] Morik, K. (1993). Balanced cooperative modeling. *Machine Learning*, 11:217 – 235.

[Morik, 1989] Morik, K. (Jan. 1989). Sloppy modeling. In Morik, K., editor, *Knowledge Representation and Organization in Machine Learning*, pages 107–134. Springer Verlag, Berlin, New York.

[Morik and Kietz, 1989] Morik, K. and Kietz, J.-U. (1989). A bootstrapping approach to conceptual clustering. In *Proc. Sixth Intern. Workshop on Machine Learning*.

[Morik et al., 1993a] Morik, K., Potamias, G., Moustakis, V., and Charissis, G. (1993a). Knowledgeable learning using MOBAL - a case study on a medical domain. In Kodratoff, Y. and Langley, P., editors, *Proceedings of the Applications Workshop at ECML*.

[Morik and Rieger, 1993] Morik, K. and Rieger, A. (1993). Learning action-oriented perceptual features for robot navigation. In Giordana, A., editor, *Learning Robots - Proceedings of ECML Workshop*.

[Morik et al., 1993b] Morik, K., Wrobel, S., Kietz, J.-U., and Emde, W. (1993b). *Knowledge Acquisition and Machine Learning - Theory, Methods, and Applications*. Academic Press, London. to appear.

[Muggleton and Feng, 1992] Muggleton, S. and Feng, C. (1992). Efficient induction of logic programs. In Muggleton, S., editor, *Inductive Logic Programming*, chapter 13, pages 281–298. Academic Press, London.

[Parsons and Puzey, 1992] Parsons, T. and Puzey, N. (1992). MLTalgorithms applied to design for manufacture - a case study. Technical Report BAE/P2154/mlt226/1, British Aerospace.

[Sommer et al., 1992] Sommer, E., Morik, K., Andre, J.-M., and Uszinsky, M. (submitted end 1992). What online machine learning can do for knowledge acquisition - a case study. *Knowledge Acquisition*.

[Turing, 1959] Turing, A. (1959). Intelligent machinery - a heretical theory. In Turing, S., editor, *Alan Turing*. Cambridge.

[Vosniadou and Brewer, 1993] Vosniadou, S. and Brewer, W. (to appear 1993). Mental models of the day/night cycle. *Cognitive Science*.

[Wirth, 1989] Wirth, R. (1989). Completing logic programs by inverse resolution. In Morik, K., editor, *Proc. Fourth European Working Session on Learning (EWSL)*, pages 239 – 250, London/San Mateo, CA. Pitman/Morgan Kaufmann.

[Wrobel, 1988] Wrobel, S. (1988). Automatic representation adjustment in an observational discovery system. In Sleeman, D., editor, *Proc. of the 3rd Europ. Working Session on Learning*, pages 253 – 262, London. Pitman.

[Wrobel, 1993] Wrobel, S. (1993). On the proper definition of minimality in specialization and theory revision. In Brazdil, P., editor, *Machine Learning - ECML-93*, pages 65 – 82. Springer, Berlin, Heidelberg, New York.

[Yokomori, 1986] Yokomori, T. (1986). Logic program forms. *New Generation Computing*, 4:305 – 319.