Knowledge Representation in Machine Learning

Filippo Neri and Lorenza Saitta

Dipartimento di Informatica, Unversità di Torino, Corso Svizzera 185, 10149
TORINO (Italy), Email: {neri,saitta}@di.unito.it

Abstract. This paper investigates the influence of knowledge representation languages on the complexity of the learning process. However, the aim of the paper is not to give a state-of-the-art account of the involved issues, but to survey the underlying ideas. Then, references will be provided only occasionally and all the specific quantitative results are left to the presentation. Finally, the paper is intentionally unbalanced, because a larger space is given to those issues that are more novel or less investigated in the literature.

1 Introduction

A large variety of approaches (e.g., symbolic, connectionist, reinforcement-based, evolutionary), of methodologies (such as inductive, deductive, abductive, analogical, case-based) and of algorithms are currently available to address the problem of building learning machines. Even if these approaches have been able to offer solutions to some interesting real problems, a large scale application of automatic learning techniques to real life has still to come. A major problem against an easy scaling up is computational complexity. Given a problem, i.e., the specification of a task (e.g., classification, control), the nature of the target knowledge (for instance, expressible in propositional or predicate logic) and the description of the environment in which learning has to take place (availability of examples, background knowledge or teacher, presence of noise), first an approach has to be chosen and, then, a method compatible with the approach, eventually implemented in a specific algorithm, has to be selected. Obviously, the task, the target knowledge and the environment have all a relevant impact on the choice of suitable approaches and algorithms. The resulting global selection exhibits a complexity which delimits the maximum size of the solvable problems. However, the choices are usually not unique and more or less wide room is left for some kind of optimisation. We are interested here, in particular, in the trade-off between reduction in the complexity of learning and quality of learned knowledge, due to issues of knowledge representation and reasoning. By complexity, we mean algorithmic complexity; complexity evaluated as minimum number of training examples is only handled as a parameter possibly affecting the preceding one. Problems of algorithm efficiency are not handled here and we will assume that the selected algorithm is as much optimised as possible. The nature of the target knowledge affects the amount of computational resources needed in more than one way. The first one is through its very nature: for instance, hypotheses expressed as first order logic formulf may have to be found in infinite search space;

moreover, some operations on them (e.g., matching), are inherently exponential [11]. Complex hypotheses (for instance, concepts with several modalities) may require an excessive number of training examples. On the other hand, reasoning mechanisms, used to learn, span a wide range of complexity, and not all of them are equally applicable to any type of target knowledge. A typical example, in this respect, is time- dependent knowledge, such as the one needed for describing the behaviour of a dynamical system. Among the several issues that arise in the analysis of the mentioned trade-off and in finding possible equilibrium points , we will concentrate here on the possible relationship between the type of formalism selected for representing the target knowledge and the means for reducing the amount of search for good hypotheses. When the solution of a learning problem (i.e., a body of knowledge) is hidden inside a very large hypothesis space, the probability of finding it or, at least, a good approximation of it, may be very small, requiring thus a large amount of computational resources. We will consider here four ways of coping with this situation.

- To reduce the size of the hypothesis space,
- To focus the search toward a subspace of the hypothesis space,
- To improve of the learning environment,
- To increase the search exploration power.

Each one of the four kinds of approaches may or may not be either applicable or effective, depending on the type of representation chosen for the target knowledge. The goal of this talk is precisely to investigate this dependency. For types of representation we intend a broad partition of representation formalisms, such as symbolic (e.g., rule sets, decision trees, logical formulf), connectionist (set of numerical parameters), bit- strings, exemplars or time-dependent functions.

2 Size Reduction of the Hypothesis Space

A first and obvious way of possibly easing the search for a hypothesis is to limit the size of the hypothesis space itself. This can be done (as usually it is) by imposing constraints on the target knowledge, aimed at limiting the expressive power of the representation language. This kind of restriction may prove effective in obtaining, for instance, polynomial learnability [as in the COLT approach], but, on the other hand, increases the probability of loosing the correct hypothesis, if this one cannot be represented in the reduced language. Still, we might want to obtain an approximate solution and to evaluate how good it is [5, 22].

An important dichotomy, in this respect, is propositional (<attribute- value> pairs descriptions) versus First Order Logic (relational) representation languages. Using a FOL language, many problems, such as, for instance, matching or testing for subsumption, become computationally intractable and even the very notion of generality may acquire more than one meaning [20, 1, 2]. As a consequence, more strict restrictions have to be imposed on the language syntax (e.g., determinate literals) or semantics (e.g., only one-to-one variable-constant unification), in order to keep computational complexity acceptable. It is not surprising, then, that relational target knowledge has not been considered, until recently, in connectionist and evolutionary approaches and that the problem of defining an adequate metrics between structured exemplars or descriptions is still an unsolved problem.

There are at least two possible ways of coping with the problem of language expressiveness: one is constructive learning, i.e., a dynamic definition of the hypothesis language, according to the needs emerging during learning. Constructive induction has been mostly addressed in symbolic approaches, even though some kinds of ANN may be considered as being able to perform it, as well, by dynamically changing their structure and not only their weights. Another way of handling the problem is to use representation languages at different levels of details (a fundamental aspect of the human thought), i.e., to use abstraction. Abstraction has been mainly used in problem solving [19], and only recently the ML community started to pay attention to this mechanism [6, 12, 7]. Notwithstanding the multiple definitions of abstraction, all their proposers agree on the intuitive meaning of abstraction as a mechanism to build up a simpler representation scheme than the one in which the problem at hand has been originally formulated.

Abstraction, dealing with transformations between representation spaces, offers a new perspective to learning, in that it addresses the fundamental dilemmas involving knowledge simplicity, meaningfulness, predictivity and task-dependency. In learning, abstraction has to be distinguished from generalisation (even though some authors have used the two terms as synonyms) and is in no way intended as an alternative mechanism to it; on the contrary, generalisation and abstraction have complementary properties and goals. Generalisation has been, and remains, the basic mechanism for searching hypotheses, whereas abstraction provides a mechanism for representing these hypotheses on a hierarchy of levels. In other words, abstraction is basically an organisational mechanism, which imposes an internal structure to the world, in such a way that a meaning can be easily associated to the component parts of the structure, reducing thus the cognitive effort for handling the world representation. As an example, it is much more difficult to associate the concept of a table to the thousands of pixels in a table picture rather than to a structure composed by some interrelated legs and top.

Then, a useful notion of abstraction in learning is one that preserves both the more-general-than relation and the extensional properties of concepts (i.e., their coverage) across hierarchical levels of representation spaces. This amounts to the fact that any hypothesis, generated inside any representation space (usually through a generalisation/specialisation process), is guaranteed to be extensionally equivalent to the same hypothesis represented in any other more or less abstract space. In other words, generalisation deals with the extensional aspect of concepts, whereas abstraction deals with the intensional one. We can summarise by saying that abstraction has to do with suitably representing hypotheses, generalisation with finding hypotheses. We do not want that uncontrolled generalisation occurs only as a side-effect of knowledge representation changes of the same hypothesis. This point appears even more clearly if we have to translate, from one level to another, also a domain theory, which should be kept semantically equivalent.

A semantic abstraction, preserving concept extension, is somewhat in contrast with most definitions proposed in AI. On the other hand, it is well on the line of the Abstract Data Types theory, used since a long time, for instance, in structured programming, in program specification and analysis and as a basic concept of object-oriented languages. The property of being at constant information is fundamental for this kind of abstraction: in fact, the semantics of a program shall be exactly the same, whatever the level of details used to describe the program may be. On the other hand, the abstract data types theory has a feature which is absent from the abstraction used so far in AI. This last, in fact, has been only concerned with changes in the language predicate set, whereas the first one builds up objects (i.e., the data types), defined in terms of properties and interactions with the external world and other objects; each object has to be addressed as a whole, disregarding its internal structure and actual implementation. The process of building up compound conceptual objects, synthesising groups of elementary pieces of information available in the ground world, and, then, hindering their internal structure in the abstract world, is the central core of the abstraction mechanism in learning. It has been frequently used in pattern recognition, especially in image analysis.

For these reasons, not only new predicates [17], but also new terms as compound objects have to be invented [7]. Building up new data types, representing intermediate concepts useful to describe higher-level ones, is the key both for obtaining meaningful, human-like concept representations and for reducing the combinatorial complexity of the learning process. Results in this sense have already been reported. The introduction of term abstraction is a key difference between the notion of abstraction sketched here and constructive induction, as it is handled, for instance, in ILP [16]. It is possible to quantitatively evaluate the reduction in search and matching obtained by introducing new compound terms both in the hypothesis language and in the (possibly available) domain theory.

Even though abstraction can be usefully applied to propositional languages to compact knowledge bases, its major role emerges in first order languages, exactly because of the strong impact on complexity due to the definition of composite objects. An interesting question is whether abstraction can play a role in connectionist approaches, by associating an individuality to specific subnets, in such a way that they could be used as building blocks to construct larger networks. And, if yes, whether this internal structuration spontaneously emerges as a consequence of increasing the size of the networks. If this would be the case, we could assist to the natural creation of symbols from the subsymbolic representation level for the sake of saving cognitive efforts.

3 Focusing the Search toward Hypothesis Subspaces

A very effective way of focusing the search in a space of hypotheses is to use a-priori knowledge, which limits the search to that subset of hypotheses that can be explained by the theory itself. This line of reasoning, started with the EBL methodology, has led, more recently, to the multistrategy approach, attempting to control complexity by using more sophisticated learning system, including different reasoning mechanisms, with the aim of better exploiting the variety of available a- priori information [13, 14, 4, 21]. Central to this kind of approach is the notion of explanation , connected, in turn, with the nature of the used knowledge and the reasoning mechanism performed to obtain it: inductive, deductive, abductive or analogical. Attempts have been done to characterise the nature of a hypothesis obtained with these methods, trying to clarify, at the same time, the philosophical foundations of learning [13]. In particular, a precise definition of the inductive/abductive nature of a hypothesis has been suggested [2]. This definition tries to capture the intuitive feeling that the only support of an inductive explanation is a supposed similarity between unobserved individuals and observed ones. In other word, an inductive hypothesis allows the validity of properties, observed on a set of individuals, to be extended to unobserved individuals, whereas an abductive one allows unobserved properties to be applied to observed individuals. Hence, the generation of an inductive hypothesis does not need a theory relating each other properties, whereas the generation of an abductive hypothesis does. The distinction between inductive and abductive hypotheses strictly parallels the dichotomy extension vs. intention. In fact, inductive hypotheses are related to (concept) extensions, whereas abductive hypotheses are related to (individual) intensions.

It can also be shown, both theoretically and experimentally, that the use of an abstract causal model of the domain, in connection with abductive reasoning, has the advantage of strongly reducing both the search in the hypothesis space and the required number of examples, keeping at the same time high the probability of finding a good hypothesis. The reason of the potentially limited complexity of abducing first causes, in comparison with a deductive approach, mostly resides in the possibility of making assumptions about the state of the world and in the possibility of using the abstract predicates occurring in the casual model to produce a skeleton of the target knowledge, to which surface details can be added later [23]. Another interesting debate, related to the nature of knowledge and the mechanisms to use it, is that between analogy [26] and case-based reasoning. In fact, the notion of analogy still deserves further clarification.

A-priori knowledge has also been used to help the designer in the definition of an ANN structure, obtaining interesting results toward the integration of symbolic and subsymbolic learning techniques [24].

4 Improvement of the Learning Process

Another way to cope with complexity is to try to exploit at the best the already available sources of information, i.e., to improve the learning process instead of improving the learning algorithm or increasing the number of training examples, This means that we can try to exploit the possible presence of a teacher (human or environmental) and this can be done in at least two ways: developing systems that interact with a human expert, seeking for his/her advise especially during the revision task [15], or to carefully select the order in which information (typically examples) are presented to the system, in order to strictly guide and speed up the hypothesis formation process.

Example ordering has receiving increasing attention [3], in the framework of incremental learning and dependency upon the presentation order of the examples is a matter of controversy. On one hand, order independence could be desirable, because training examples can be chosen more freely, there is no need of backtracking and there is a smaller danger of overfitting. On the other hand, we experience, in human learning, that a suitable presentation order of selected examples can help the learner to quickly focus on the important aspects of the matter, generating thus a robust kernel of knowledge, to which border cases and exceptions can be easily added later. However, performed experiments and theoretical computations (using Gold's paradigm) show that the number of training examples required to attain the same level of performance can be drastically reduced by taking order effects into accounts [18].

The study of order effects are particularly relevant in ANN, in connection with the problem of forgetting.

5 Increase of the Search Exploration Power

If we do not have sufficient a-priori knowledge and help, then we have to rely on search. To this aim, genetic algorithms offer a powerful, domain-independent method: they have been first used in machine learning associated to Holland's classifier model, but have also been exploited in other frameworks, for instance, to train neural nets instead of the classical back propagation algorithm.

Recently, also the symbolic machine learning approach took advantage of genetic algorithms for concept induction in propositional calculus [25]. From these first experiments, genetic algorithms proved to be an appealing alternative to classical search algorithms, because of their exploration power and their suitability to exploit massive parallelism.

Recently, the extension of the genetic search to concept descriptions in First Order Logic [8, 9] greatly extended the potential of this approach. Moreover, it has been possible to extend the method to learn disjunctive concepts, by proposing a new model of evolution under the selection operator. A theoretical study, involving the determination of the system's asymptotic behaviour, has shown that the new model leads to an equilibrium state between the alternative disjuncts, which, hence, will not disappear, if the population is sufficiently large. Realistic bounds for the cardinality of the population have been derived. With the same technique, also the classical model of selection and the one of Goldberg's sharing functions method have been analysed. The results are that in the classical model only the best disjunct will survive, however large the population is. The sharing functions also show a non trivial equilibrium point, but only under given conditions. Moreover, the evaluation of the shared fitness is quadratic with the cardinality of the population, whereas the same evaluation in the new model is only linear. An extensive experimentation confirms the theoretical results. These results are compared with those asymptotically obtainable from symbolic learning algorithms (such as FOIL) and ANN [10]. Comments on the impact of parallelism will also be done.

6 Conclusions

The talk will give a comparison among techniques usable to reduce the computational complexity of the learning process (intended in a broad sense), in connection with their suitability in dependence of the representation formalism used for the target knowledge. Quantitative results are given, where appropriate and possible.

References

1. Buntine W. (1988). Generalized Subsumption and Its Applications to Induction and Redundancy, Artificial Intelligence, 36, 149-176.
2. Console L., Saitta L. (1992). Abduction, Induction and Inverse Resolution, Proc. First Compulog-Net Workshop on Logic Programming in AI (London, UK).
3. Cournejols A. (Ed.). AAAI Spring Symposium on Order Effects in Incremental Learning (Stanford, CA, 1993).
4. De Raedt, L., Bruynooghe, M. (1991): CLINT: A Multistrategy Interactive Concept Learner and Theory Revision System. Proc. First International Workshop on Multistrategy Learning (Harpers Ferry, WV), pp. 175-190.
5. Devroye L. (1988): Automatic Pattern Recognition: A Study of the Probability of Error . IEEE Trans. on Pattern Analysis and Machine Intelligence, 10 : 530-543.
6. Drastal G., Czako G., Raatz S. (1989): "Induction in an Abstraction Space", Proc. IJCAI-89, (Detroit, MI), pp. 708-712.
7. Giordana A. & Saitta L. (1990). Abstraction: a General Framework for Learning", Working Notes of the Workshop on Automated Generation of Approximations and Abstractions (Boston, MA, 1990), pp. 245-256.
8. Giordana A., Sale C.(1992). Genetic Algorithms for Learning Relations. Proc. Int. Conf. on MAchine Learning (Aberdeen, UK).
9. Giordana A., Saitta L. (1993). Learning Relations using Genetic Algorithms. In Michalski R. & Tecuci G. (Eds.). Proc. Multistrategy Learning Workshop (Harpers Ferry, VA).
10. Giordana A., Saitta L. (1994). Learning Multimodal Relational Concepts using Genetic Algorithms: A new Theoretical Model and Experimention. Tech. Rep. TR 94-3, Dip. Informatica, Torino.
11. Haussler, D. (1989): Learning Conjunctive Concepts in Structural Domains . Machine Learning, 4 , 7-40.
12. Knoblock C. (1989). Learning Hierarchies of Abstraction Spaces. Proc. 6th Int. Workshop on Machine Learning (Ithaca, NY).

13. Michalski R. (1991): Inferential Learning Theory as a Basis for Multistrategy Task-Adaptive Learning . Proc. First International Workshop on Multistrategy Learning (Harpers Ferry, WV), pp. 3- 18).

14. Michalski R., Tecuci G. (Eds.). Proc. Multistrategy Learning Workshop (Harpers Ferry, VA, 1991, 1993).

15. Morik K. (1987): Acquiring Domain Models . Int. Journal of Man- Machine Studies, 26 , 93-104.

16. Muggleton S. (1991). Inductive Logic Programming, New Generation Computing, 8, 295-318.

17. Muggleton S., Buntine W. (1988). Machine Invention of First-Order Predicates by Inverting Resolution. Proc. 5th Int. Conf. on Machine Learning (Ann Arbor, MI), pp.339-352.

18. Neri F. & Saitta L. (1993). Exploiting Example Selection and Ordering to Speed-Up Learning. In A. Cournejols (Ed.), AAAI Spring Symposium on Order Effects in Incremental Learning (Stanford, CA, 1993).

19. Plaisted D. (1981). Theorem Proving with Abstraction. Artificial Intelligence, 16, 47-108 (1981).

20. Plotkin G. (1970). A Note on Inductive Generalization. Machine Intelligence, 5, 153-163.

21. Saitta L. (Ed.). Notes of the ML-Net Multistrategy Learning Workshop (Blanes, Spain, 1993).

22. Saitta L., Bergadano F. (1993). Pattern Recognition and Valiant's Learning Framework. IEEE Trans. Pattern Analysis and Machine Intelligence.

23. Saitta L., Botta M. & Neri F. (1993). Multi-Strategy Learning and Theory Revision , Machine Learning. 11 (2/3).

24. Towell G., Shavlik J. (1991):Refining Symbolic Knowledge using Neural Networks , Proc. Workshop on Multi-Strategy Learning (Harpers Ferry, WV), pp. 257-272.

25. Vafaie H., De Jong K. (1991): Improving the Performance of a Rule Induction System using Genetic Algorithms , Proc. Workshop on Multi-Strategy Learning (Harpers Ferry, WV), pp. 305-315.

26. Veloso M., Carbonell J. (1991): Learning by Analogical Replay in PRODIGY: First Results , Proc. EWSL-91 (Porto, Portogallo), pp. 375-390.