Explainable Induction with an Imperfect Qualitative Model

Stan Matwin **Thierry Rouget**
Ottawa Machine Learning Group
Department of Computer Science
University of Ottawa
Ottawa, Canada K1N 6N5
tel. +1 613 562 5800 ext. 6679 fax: +1 613 562 5187

stan@csi.uottawa.ca trouget@csi.uottawa.ca

Abstract

This paper addresses the problem of learning concept descriptions that are interpretable, or explainable. Explainability is understood as the ability to justify the learned concept in terms of the existing background knowledge. The starting point for the work was an existing system that would induce only fully explainable rules. The system performed well when the model used during induction was complete and correct. In practice, however, models are likely to be incomplete and incorrect. We report here a new approach that achieves explainability with models that are incomplete or inconsistent, or both. The basis of the system is the standard inductive search driven by an accuracy-oriented heuristic, biased towards rule explainability. The bias is abandoned when there is heuristic evidence that a significant loss of accuracy results from constraining the search to explainable rules only. The user can express their relative preference for accuracy vs. explainability. Experiments with the system indicate that, even with a partially incomplete and/or incorrect model, insisting on explainability results in only a small loss of accuracy. We also show how the new approach described can repair a faulty model using evidence derived from data during induction.

1. Introduction

"Classical" inductive systems [Quinlan 93], [Michalski et al. 86], [Clark and Niblett 85] rely on data, and on data alone, to generate the learned concept description. For the last several years, however, there is a growing focus in the machine learning community on interpretability of the inductively learned rules. Indeed, many believe [Aha and Riddle 95, Bratko 95, Marchand 95, Evans and Fisher 93] that interpretability of the results is essential to the acceptance of the learned knowledge by the user, as well as to its correct application. An obvious way of ensuring that interpretable rules are learned is to impose that they should be *explainable* by some pre-existing knowledge in the application domain. Where does this knowledge come from? Any significant, real-life learning application is preceded by a period of time in which the 'knowledge engineers' familiarize themselves with the basic knowledge of the application domain. Acquisition of this knowledge is necessary in order to understand the problem and its attributes, as well as for communication with the domain experts. If the domain knowledge is coded in some runnable representation, it could be used, after induction has been completed, to filter out rules that cannot be explained (in some precise sense) by that knowledge. This approach, however, does not exploit the use of knowledge *within* the learner. We are interested in ensuring explainability of learned rules by involving the knowledge in the search process that constitutes the basis of induction. This is the context of the work presented here.

Surprisingly little has been done in the area of incorporating the previous knowledge inside a learning algorithm. Notable exception is the work of [Ortega and Fisher 95]. The focus there, however, was on exploiting domain knowledge for accuracy, rather than explainability of the results of learning.

We decided to work on obtaining explainable and accurate rules that classify concepts related to operations of systems for which there are qualitative models in the sense of [Forbus 84], [Kuipers 95]. We use a qualitative model to represent the system under study and to guide an inductive search toward rules which are explainable with respect to this qualitative model. Our earlier work [Clark and Matwin, 1993a, Clark and Matwin, 1993b] described the first attempt at learning explainable rules in the presence of a qualitative model. In this early work, explainable rules are exhaustively extracted from the model and, during induction, the choices available to the search engine are restricted so that only explainable rules are selected. We refer to this approach as *model-only biased search* (MOBS), as the standard inductive search driven by an accuracy-oriented heuristic is biased only by the model. While obtaining total explainability of the induced rules, this approach has one major limitation: it assumes that the qualitative model is reasonably complete and correct. The extension that we present in this paper, *model-and-data biased search* (MDBS) is intended to cope with this limitation of the background knowledge. We show here how knowledge which is either incomplete or incorrect or both, can still be useful in learning explainable rules. If the model is severely imperfect, it might turn out that restricting the search to explainable rules only leads to poorly accurate rules. The search task in our current work is thus to try to maximize explainability while maintaining accuracy. As a result, the new method may learn rules which are not totally explainable, but still represent the level of explainability substantially higher than the default one (i.e. *a posteriori* explainabilty of the rules induced by an inductive learner where background knowledge is not used during search).

Finally, when data somewhat conflicts with the qualitative model, one may want to revise the model according to overwhelming statistical evidence found in the training data. The method that we propose here can be used to correct or to complete an imperfect model.

2. Knowledge representation

This research can be viewed as developing a special case of the theory-guided learning paradigm, in which the domain theory is a qualitative model of the studied physical system and the learning is rule induction from data. Given the state of a physical system at a given time T, we want to predict the evolution of a particular target parameter during the next time step, i.e. whether its value will have increased or decreased by time T+ΔT.

In our research, we use qualitative models to represent expert knowledge of the physical system under study. While very similar to models of the Qualitative Process Theory [Forbus, 84], one should note one essential difference: our models are incompletely specified. No qualitative range of values for parameters (qualitative spaces) is defined, so we are unable to solve conflicting influences during simulation. Thus, the main difference is that our models on their own cannot be used for simulation or prediction. Instead, their role is to describe which processes we expect to take place in the physical system and to constrain induction to rules we can explain with respect to these processes.

The qualitative model is a qualitative description of the behaviour of the physical system. This description is a graph whose nodes are parameters of the application domain and whose arcs represent relationships between parameters (arrows indicating temporal precedence). Our qualitative models use six elementary types of relationships between its parameters.

The Q+ (qualitatively proportional) relationship stands for: "there exists a function f that determines Y, and is monotonically increasing in its dependence on X". The I+ relationship means that "X is a cause for Y to rise" or that the rate of change of Y (dY/dt) varies monotonically with X. Thus, it means that if X exceeds a given value then Y increases (or will increase). In the following, "X is high" actually refers to "X is greater than a given value". Thus, "X is high" should not be interpreted as "X's value is high in the range of X's possible values". This simplification is introduced to make qualitative explanations of the system's behaviour shorter. I and Q links are essentially different. If two parameters X and Y are connected by a Q+ link, knowing the value of X gives us information about the value of Y.. Furthermore, if we know that X is increasing, then we also know that Y is increasing. However, knowing X's value does not give us any information on Y's future change. For an I+ link, knowing the value of X tells us whether Y is increasing or decreasing, but it does not give us any information on Y's current value.

In our qualitative models, we will also use a third label I* in addition to I and Q to denote a self-stabilizing feedback loop. This link is not included in Forbus' Qualitative Process Theory. An I*+ link means that if X is high, then Y will rise until it reaches a new constant value. In the same manner, an I*- relationship denotes an inverse monotonic relationship. For each of the links Q+, I+, I*+, there is a symmetric link Q-, I-, I*- which represents a decreasing change of the second argument with the increase, or with a high value of the first argument.

Fig. 1 shows an example of a system used throughout this paper, and Fig. 2 shows the qualitative model of this system. The system consists of a number of water tanks connected by pipes with valves[1]. Water enters the system through the upper pipe and into the first water tank T1. From tank T1, water may flow into T2, T3 or T4 and so on. In addition, there are control valves which may prevent water from flowing through some pipes (for example, there is a control valve on the upper pipe). These valves are controlled by the level of the tanks. For instance, the control valve on the pipe between T1 and T3 is closed whenever the level in tank T5 becomes greater than a given value, but remains open when T5's level is below this value. These control valves are means of feedback from a tank to the water flowing into it (more or less directly).

[1] this is the essentially the same application we have already used in [Clark and Matwin 93]; we have developed elsewhere two other, more complex applications that take more space to describe [Thesis, Journal Paper]

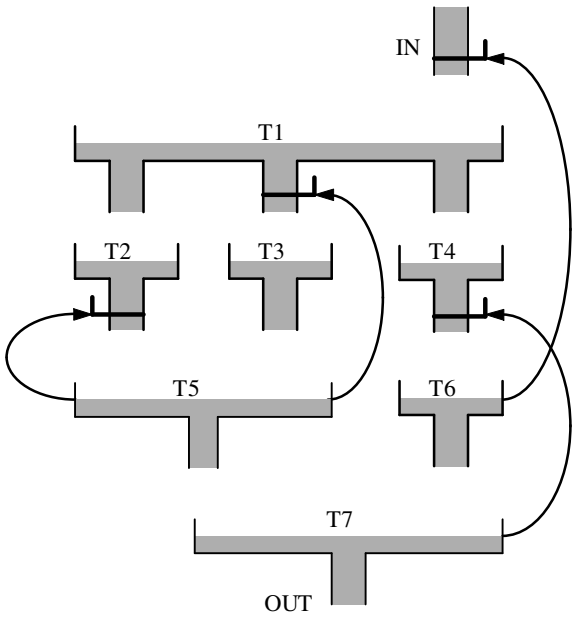

Fig. 1. The water tank system

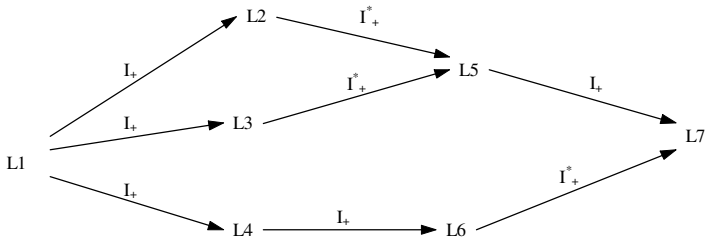

Fig. 2. Qualitative model of the system in Fig. 2 with feedback loops summarized

The only user-controllable parameter is the water flow entering the system through the upper pipe. The parameter of interest is the level of the water tank T7 (the tank at the bottom of the circuit). The level of each water tank is observable so the learning task is, given the levels L1, L2, ..., L7 of the tanks at time T, to predict whether the level L7 will have increased or decreased by time T+ΔT.
The qualitative model was based on the fluid dynamics law, which states that the flow of water out of a tank is increasing monotonically with the tank's water level. In this model, the level in a water tank increases with the level of water of a tank just above, as the out-flow from the latter also increases with its level. Thus, if the level in tank 3 is high, then the flow of water from tank 3 into tank 5 will be high and will cause the level in tank 5 to rise. [Clark and Matwin 1993] contains a more detailed description of this model. The actual model used in the current paper is shown in Fig. 2.

The rules learned by our inductive tool have the following structure:

$$if\ T_1\ and\ ...\ and\ T_n\ then\ C$$

where T_i (for i between 1 and n) is a test on some observable parameter P_j, i.e. $P_j<k_i$ or $P_j>k_i$ where k_i is a real constant value, and where C asserts either "*Pconc will increase*" or "*Pconc will decrease*" during the next time-step. Pconc does not have to be observable and can be any parameter of the application domain. However, we are only interested in learning rules predicting the change of the target parameter. We also define a **rule schema** as being a rule whose constant values are not specified. We refer to [Thesis] for other details of knowledge representation, as well as for some of the implementation details. It suffices to say here that we pre-compile the model into a list of all the possible rule schema. This database of rule schema is then used to prune, during the search, the rules derived from the data that are not covered by any schema, and are then deemed unexplained by the model.

Independently of the model, a simple numeric system written in Prolog was used to simulate the system's behavior with time and to generate data for learning.

3. Model-only Biased Search

The inductive algorithm we use is the latest version of CN2 [Clark and Niblett, 1989; Clark and Boswell, 1991]. It induces an ordered list of classification rules from examples, by relying on Laplace's error estimate as its search heuristic. Given a set of training example, CN2 performs a hill-climbing search to find the rule which covers best the examples (i.e. which covers many positive examples and not many negative examples). This search is general to specific a beam-search: CN2 starts with a rule containing no attributes test and predicting the class (for which we try to find a rule) and specializes this empty rule by adding any possible attributes tests. Then, the best specializations (according to the search heuristic) are kept to be specialized further, until no specialization is possible anymore. The best rule found during the process is added to the ruleset and the positive examples it covers are removed from the set of training examples. The process terminates when there are no positive examples to cover, or when the coverage is not significantly improved by the last rule found, or when a preset limit of the depth of the search is reached. In our learning task there are two classes (target will increase/target will decrease) and attribute tests are the parameter tests of the previous section.

In [Clark and Matwin, 1993], a modified version of CN2 taking a qualitative model into account was presented. Unlike the standard CN2, which considers any possible specialization of the candidate rules in the beam, this Model-Only Biased Search restricts the search to the subspace of rules which are explainable with respect to a qualitative model of the system. There are two main objectives of this approach. The first one is to induce totally explainable rules and thus to maximize the interpretability of the learned knowledge. The second one is to prune out rules which, by chance, perform well on the training data and which are not explainable, as these are expected to have a poor predictive accuracy on unseen data.

In order to constrain the search to the subspace of explainable rules, the specialization operation was modified. We have seen in the previous section how we could generate the list of all the rule schemata explainable with respect to a given qualitative model. This list is stored in a database in the following format: each entry is addressed by a set of parameter tests (i.e. P3 > _, P5 < _). Each entry contains its class (increase/decrease) and the list of specializations (additional parameter tests) that are allowed so that if any of them is added to the first set of tests, the rule containing the specialized set of tests and predicting the class is explainable. Note that this is obtained by processing the set of explainable rules extracted from the model and that it has to be done only once per model. Then, during induction, rather than trying all possible specializations of a candidate rule, the modified CN2 retrieves from the database the 'explainable' specializations and considers only these. By construction (since only the attribute tests which keep the rules explainable are added), the set of learned rules will contain only explainable rules.

In [Clark and Matwin, 1993], experiments conducted with MOBS for two application domains (the water tank system and the ore grinding circuit) showed that, by using the proposed qualitative models, total explainability of the learned ruleset was obtained at no cost on the classificational accuracy. As a matter of fact, a noticeable improvement of accuracy was observed, indicating that useful knowledge was indeed injected into the learning process. However the benefits on accuracy of this method depend on the quality of the model. In sequel, we propose extensions to MOBS which cope with imperfections in the model. We will also see how to revise an imperfect model and how uncertainty can be advantageously represented in the qualitative model itself.

4. Model-and-Data Biased Search

One disadvantage of MOBS, which learns exclusively explainable rules, consists in the fact that its performance depends on the model's quality. If the model is strongly imperfect (incomplete or incorrect or both), then the accuracy of the learned rules will be poor, as opposed to the accuracy of rules learned by the standard CN2 using no background knowledge.

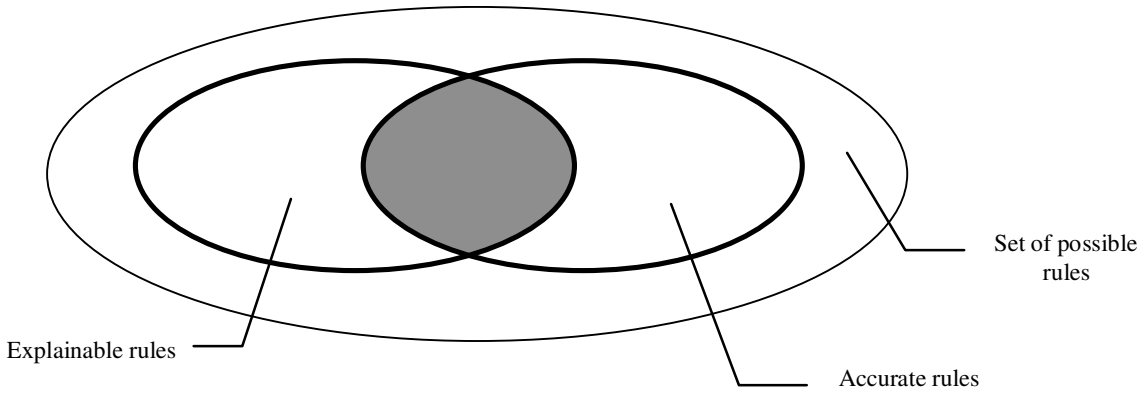

Figure 3 Intuitive view of the search spaces

Figure 3 gives an intuitive perception of the problem. The standard CN2 algorithm heuristically searches in the space of rules for rules which have a high predictive accuracy. We thus expect rules induced by the standard CN2 to belong to the set of accurate rules. MOBS described in the previous section looks for rules which also classify well the training examples, but only in the subset of rules which are explainable with respect to the qualitative model. The resulting rules will probably belong to the intersection of the two subsets, i.e. the darkened area. We can then distinguish two extreme situations. On the one hand, if the model is good, then the overlap between the set of explainable rules and the set of rules which classify well the data is important (ideally the two sets matches exactly) and induced rules are both explainable and accurate. On the other hand, if our model is very deficient (incomplete, incorrect or both), the overlap between the two sets will be small or empty. This is a direct consequence of the fact that the domain knowledge expressed in the model is not accurate with respect to the physical system and thus with the examples. In this case, the standard CN2 will produce accurate rules but which are poorly explainable (CN2 is not biased for explainability at all and produces poorly explainable rules even when the model is perfect). If we resort to MOBS, the induced rules will be totally explainable but will have a poor predictive accuracy since, to cover as many training examples as possible, search will probably extend to the whole set of explainable rules (and not only the darkened area). Neither alternative is really satisfactory. In this section, we propose a strategy which first searches for a rule in the intersection of the set of explainable rules and the set of accurate rules (the darkened area). If the result of this search is not satisfactory (i.e. explainable rules are not accurate enough), then the search expands to the whole search space. This approach biases the search first by the model. Then, when the model fails to ensure the required accuracy, the search is driven by the data. Hence the name *Model-and-Data Biased Search* (MDBS).

As described in section 2, explainable rules are exhaustively extracted from the model and stored in a database in such a way that, given an explainable candidate rule, the corresponding entry lists the specializations which keep this rule explainable. Thus, whether we generate explainable or non explainable rules is determined by whether we rely exclusively on the explainable specializations or whether we allow other specializations to be considered during the inductive search. From this point of view, CN2 and MOBS represent two extremes: CN2 tries all the possible specializations of a rule, not caring whether they are explainable or not, and MOBS tries only explainable specializations.

In the case where the qualitative model is imperfect, relying exclusively on explainable specializations will lead to poor results. However, we still want to exploit the parts of the model which are correct. The solution we adopted is to constrain the search to explainable rules (and thus to rely on the qualitative model) unless there is some evidence that induced rules perform poorly. If such a situation is detected during the induction of a particular rule, then we give up the model and rely on data exclusively until the end of this rule induction. The difficulty in this approach is to evaluate in a objective manner whether a given rule performs poorly. Obviously, this evaluation has to be done only on training data. We decided to rely on Laplace expected accuracy, which estimates what the predictive accuracy of a candidate rule will be on unseen data and which is also CN2's search heuristic. [Clark and Boswell, 1991] discuss the advantages of using the Laplace expected accuracy in rule induction, instead of the information gain measure [C4.5].

During induction of a particular rule, only explainable specializations are considered at first. If at least one specialization has an expected accuracy above a user-defined value called the **threshold**, then we keep the best explainable specializations in the search beam. Thus, if at every depth, at least one of the explainable specializations passes the threshold test, the search will produce the same results as MODS. However, if, at a given depth, none of the explainable specializations in the current beam has an expected accuracy higher than the threshold, then we give-up the current specializations and backtrack to the previous depth. Further operations are not constrained by the explainability requirement, i.e. we try all the specializations, no matter if they are explainable or not. At this point, the algorithm indeed behaves like the standard CN2 algorithm, and does not take the background knowledge into account until the end of the execution of the rule induction.

Giving up the explainability requirement is actually an expansion of the search space from the space of explainable rules to the space of all possible rules. We allow this search expansion to take place when explainable rules seem to perform poorly, because it suggests that the qualitative model does not explain adequately the set of uncovered positive examples. In this case, we prefer to rely on pure induction to learn the corresponding rule or portion of rule. There is also a second situation in which the search expansion takes place, namely when the set of explainable specializations of the rules in the search beam is empty. Such a situation can result from an incompleteness of the model and thus, we allow the search to expand to the set of all possible rules to check whether there exists some non explainable specializations of the candidate rules in the beam with an expected accuracy higher than the last explainable rule before the search expansion.

Similarly to CN2 (and also to MOBS), MDBS executes an hill-climbing search, keeping the best candidate rules in the search beam to be specialized further. In MDBS, MOBS's bias for explainable rules can be given up during search, according to some evidence found in the data that the model does not have the expected quality. The value of the threshold under which search is expanded is a user-defined parameter. It reflects how confident the user feels in the qualitative model, as opposed to statistical evidence found in training data. The lower the threshold, the more we tend to trust explainable rules extracted from the model.

If the threshold value is low (close to 0), then we impose a low minimum expected accuracy on the explainable rules. As a consequence, search expansions will be rare, since the candidate explainable specializations will be most of the time be evaluated as good enough. Thus we will have highly explainable rules. Explainability is favored against accuracy.

If the threshold value is high (close to 1), we impose a severe constraint on the expected accuracy of the explainable rules. As a result, search expansions will take place more often, resulting in potentially less explainable (but more accurate) rules. Accuracy is favored against explainability (see Fig. 4).

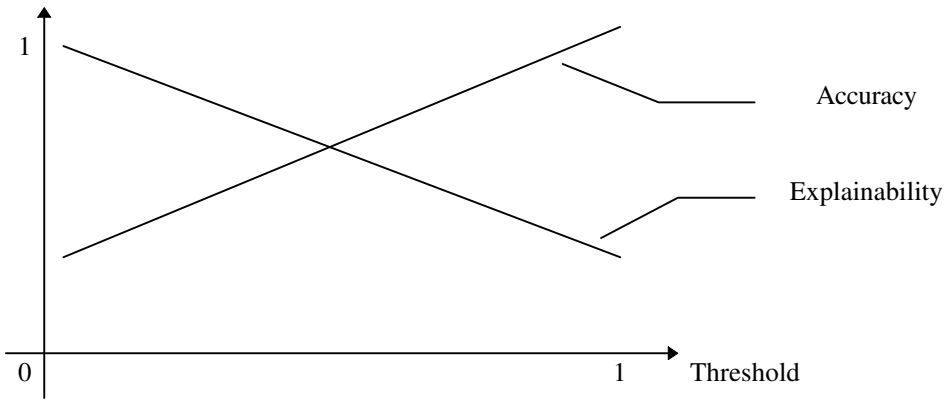

Fig. 4. Qualitative variations of explainability and accuracy with the threshold value

In fact, the threshold can be seen as the strength of the bias for accuracy as opposed to the bias for explainability.

5. Experimental results

In this section, we present an experimental evaluation of our model and data driven search. We compared it to CN2 (with no background knowledge) and MOBS. Experiments were conducted in three application domains, the water tank system and the ore grinding circuit, both already introduced in [Clark and Matwin, 1993], and the analog electric circuit domain [Rouget 95]. Because of limited space, and since experimentals results in the three domains are very similar, we will only present the results obtained in the water tank system (however, any aspect specific to one domain only will be mentioned). The complete results are available in [Rouget 95]. In the experiments desrcibed here, we have investigated the influence of the threshold parameter of MDBS on the accuracy and the explainability of MDBS. We have performed a comparative analysis of MDBS's behaviour, in terms of the above characteristics, with the behaviour of MOBS and CN2. All the experiments have been performed using 280 training and 120 testing examples. The example set was collected by running a simulator of the system (totally independent of the qualitative model), initialized to a random state, and then running for 1 sec. CPU time. The beam width in the search was 5, and depth limit was 3, 4, and 5 (the results show average values for these three depths) for the complete model, and 3 for the incomplete model. All the results are the averages of 20 runs. Two sets of experiments were performed. In the Figs. 4 and 5 we show the behaviour of all three learners in the presence of a complete model.

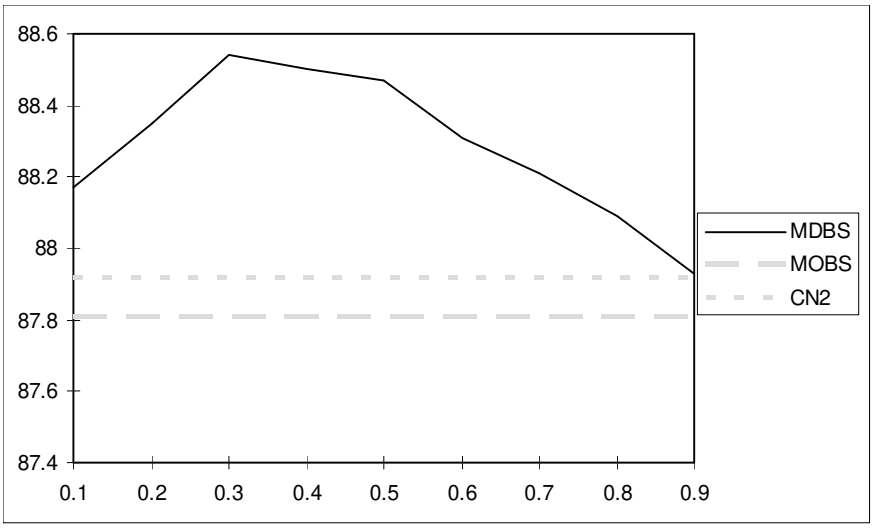

Figure 5. Accuracy variation with the threshold value

Fig. 5 shows the accuracy of MDBS, changing with the change of the threshold. Clearly, the accuracies of CN2 and MOBS are independent of the threshold. CN2, MOBS and MDBS do not differ considerably: all three are within 1% of each other. It is significant, however, that the accuracy of MDBS is no less than the accuracies of the two other. The fact that the accuracy of CN2 is the lowest among the two methods could be attributed to the fact that the model eliminates the rules which represent spurious regularities in the learning data, unlikely to occur outside the training set.

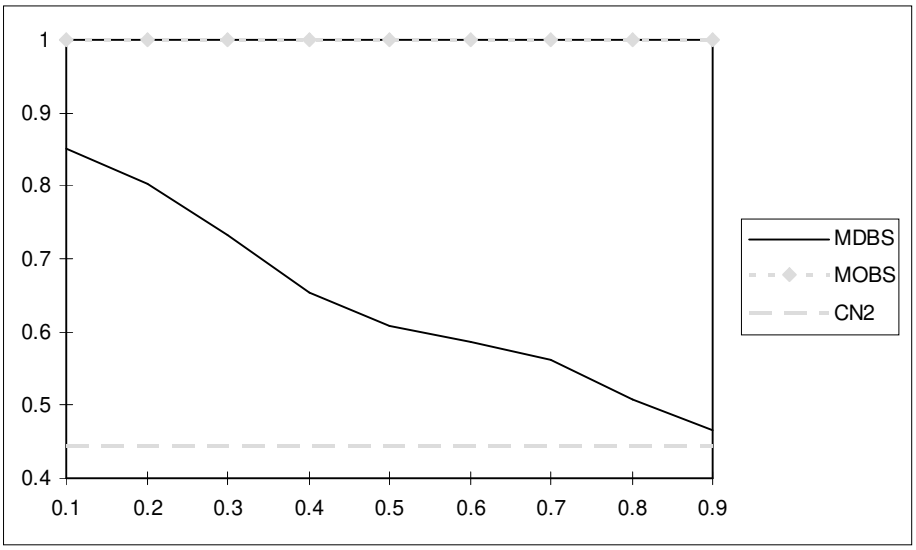

Figure 6. Explainability variation with the threshold value

Fig. 6 shows the influence of the changing threshold on the explainability of the rulesets learned. Explainability is the percentage of rules induced that can be explained by the model in the sense given in sec. 3. Clearly, explainability of CN2 rules remains constant (and fairly low), while explainability of MOBS rules is - by the system's definition - 100%.

The learning time of MDBS grows with the increasing threshold. The larger the threshold, the fewer explainable rules will be more accurate than the threshold, and the more likely the search will expand

beyond the explainable rules. Any such expansion represents time overhead with respect to theb base case of MDBS, in which no search expansions occur[2].

Figs. 7 and 8 show the behavior of the three systems in the presence of an incomplete model. Incompleteness was introduced by removing from the initial model shown in Fig. 2 the link I^*+ connecting L_6 and L_7. Fig.7 shows how the accuracy of MDBS changes with the changing threshold, again with the constant accuracy of CN2 and MOBS. As expected, accuracy of MOBS is inferior to the accuracy of CN2, as this time the model is incomplete, and MOBS will inevitably reject some well-performing rules because of their inexplainability. The accuracy of MDBS increases with the growing threshold, as the system is more and more likely to expand the search beyond the rules explainable by the incomplete model, and cover more data. It is interesting to observe that the accuracy is significantly better than in MOBS. In this experiment, the accuracy gain seems to be traded off against decreasing explainability (see Fig. 8).

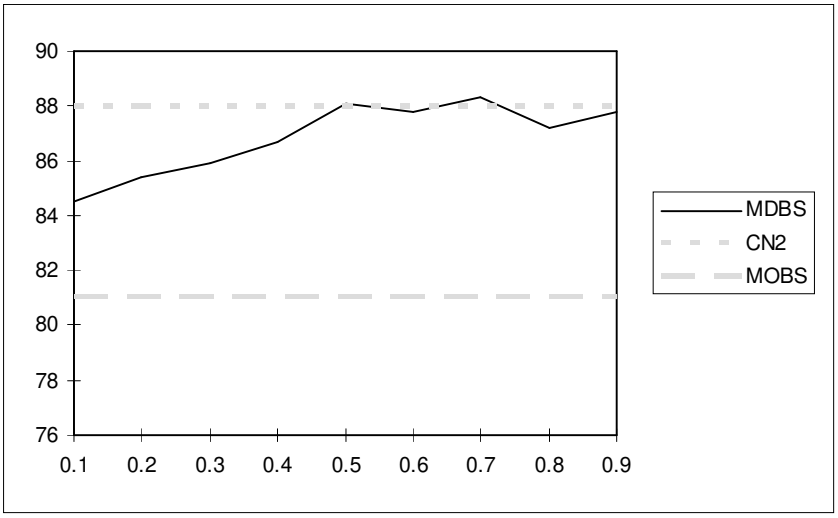

Figure 7. Accuracy variation with the threshold value (incomplete model)

As it could be expected, explainability in the presence of an incomplete model decreases with the growing threshold. As the system emphasizes more and more the role of accuracy (large threshold), search expansion occurs more often, resulting in the decrease in explainability.

[2] this overhead could be considerably reduced in a future implementation with the use of a more efficient search expansion algorithm

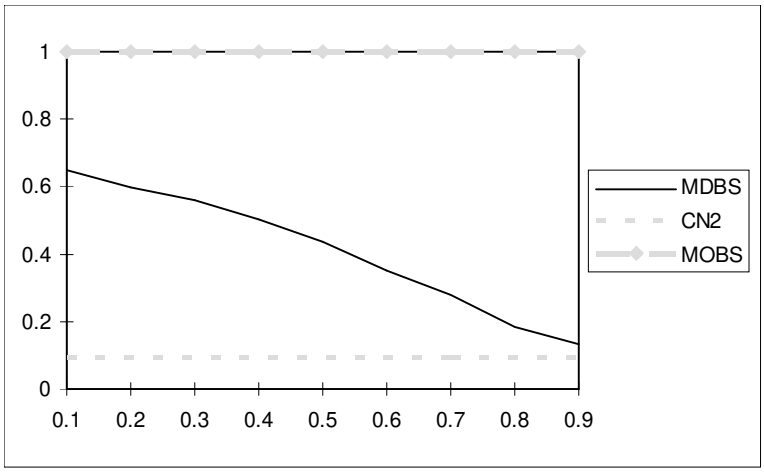

Figure 7 Explainability variation with the threshold value (incomplete model)

We have also performed a similar set of experiments with a model that was incorrect, rather than incomplete. The results were very similar for the results for an incomplete model discussed above. This indicates robustness of the MDBS approach.

5. Analysis and Discussion

In general, experimental results suggest that the performance of the method is good when the quality of the model is high. Small increase in accuracy is due to the noise-resistance of the knowledge-based inductive search. With the deteriorating model, the MDBS accuracy is superior to MOBS, and the trend seems to indicate that explainability is traded-off for accuracy. Closer inspection of the results in Figs. 7 and 8 indicates, however, that the trade-off is heavily biased in favor of explainability. For instance, for the value of the threshold = 0.2 the accuracy of MDBS is approximately 85%, while the explainability is approximately 60%, while for the value of the threshold = 0.7 the accuracy is 88% while the explainability is approximately 30%. A small sacrifice of 3% in accuracy experienced with the change of the threshold from 0.7 to 0.2, inconsequential in many applications, results in a considerable increase of 30% in explainability. Experiments in two other domains have confirmed this trend of MDBS: a small loss in accuracy results in a considerable gain in explainability.

The results of sec. 5 indicate that, firstly explainable learning with an imperfect model is still interesting, as long as one can parameterize the learning process with one's confidence in the model (this is the role of the threshold). Secondly, the expected trade-off between accuracy and explainability is not really a trade-off, but an exchange of a small loss in accuracy for a big gain in explainability. This is achieved by controlling the MDBS with the threshold parameter.

The MDBS approach can be also used to repair the imperfect model. When a model is faulty (incomplete or incorrect), we expect search expansion to take place to induce rules corresponding to the missing or incorrect knowledge. As a result, we will learn rules which cannot be explained by the qualitative model, since they do not correspond to any combination of pairs of fully specified rules extracted from the model. These non-explainable rules can be exploited to help revisions of imperfect models, by pointing out what the weaknesses in the model are. In order to collect this information, we proceed as follows:

- each time a search expansion occurs and a non-explainable specialization of the candidate rules in the beam is selected as the best rule so far (i.e. whenever a search expansion actually produces a non-explainable rule with a better expected accuracy than explainable ones), we store the rule schema corresponding to this rule in a file. For example, if the best rule in the beam[3] after a search expansion is:

if p1>45 and p2<135 and p3>-225 then target will increase

and if this rule is not explainable, the rule schema below is stored:

if p1>_ and p2<_ and p3>_ then target will increase

The interest of storing the rule schema rather than the rule itself comes from the fact that we are not interested by the numeric values but rather by the form of the rule itself. The rule schema is all we need to compare the rule against the qualitative model for possible revisions. Moreover, two rules which differ only in their numerical values contain the same information, from the point of view of explainability. In this case, storing the rule schemas will clearly indicate their similarity. Once the induction is over, all rule schemata stored are normalized. The normalization consists only in inverting the rule schemata predicting a decrease of the target parameter with an increasing (resp. decreasing) observable, so that they also predict an increase of the target with the decrease (resp. increase) of this observable. The list of normalized rule schemata is ordered from the one which occurred most often to the one which occurred the least often. By examining this list, it is now possible to revise an imperfect model.

- if the model is incomplete (an important link is missing in the model), then we expect the non explainable rule schema induced most often to be the rule schema corresponding to this missing knowledge (corresponding to the missing link).

- if the model is incorrect (a link in contradiction with the actual behaviour of the physical system is present in the qualitative model), then we expect the non explainable rule schema induced most often to contradict some part of the model.

Based on the information generated during induction, the expert in charge of revising the model can add links corresponding to the most often induced non explainable schemata. If this additions explicitly contradicts some links already in the model, then the expert should consider the correction of this part of the model, rather than a simple addition. We have experimented with this approach to model revision by removing the link I^*+ connecting L_6 and L_7 from the initial model in Fig. 2, and then performing the semi-automatic process described above. The resulting I adding the missing link to the model, thereby revising it correctly. However, there are some limitations in the possible revisions due to the nature of our background knowledge. A rule schema does not actually correspond to only one link in the qualitative model (because of the effect propagation and of the operationalization process). Indeed, the only information given by a non explainable rule schema is: *here is a parameter (possibly more than one) and its relationship to the target class as suggested by the data. The influence of this parameter on the target cannot be explained by the qualitative model as it is*. We could then automatically complete the model by adding a link from this parameter to the target and which corresponds to the unexplained influence (and if this link contradicts a previously existing link, we correct the latter rather than doing an addition). However, this kind of revision would be a simplification of how an expert would revise the model, in the sense that it does not include any other parameter and that it does not try to use an already existing path to the target.

[3] Note that we store the best rule so far right after the search expansion and only if it is not explainable. We are not talking here of the rule which will finally be selected when the search is over. As a consequence, the stored rule may differ from the rule produced by the whole search.

5. Conclusion and future Work

In this paper, we have described a system that learns explainable rules in the presence of imperfect background knowledge. This is an important advance over previous work, which was geared towards learning using background knowledge which is complete and correct. In any practical application, however, the available and symbolically represented domain knowledge is bound to be incomplete and/or incorrect. The Model and Data Biased Search approach induced explainable rules as long as the interpretability constraint does not hamper the accuracy too much. Once the explainable rules are unable to achieve the accuracy level set by the user, the search expands towards rules that, although not explainable, yield high accuracy. We have implemented this approach, adapting CN2 as our basic inductive learner and using the Qualitative Physics models of devices as the background knowledge. Experimented in three different domains indicate that the MDBS approach is superior to the previous work when the model is incomplete. We show that in this situation often allowing a small accuracy loss results in a considerable gain in explainability. Finally we show how the learning method that we have developed is used in the process of revision of an incomplete of incorrect model. This last - and, to the best of our knowledge, innovative - aspect of our work may lead to a fruitful connection between Qualitative Modelling and Machine Learning.

Future work will involve an approach that will allow the user to express their uncertainty about the model with a finer grain than in the MDBS method. Belief weights, associated with the links in the model will provide a representation of uncertainty of just the parts of the knowledge at the model design level. During induction, these weights will be used to determine the belief value of candidate rules (non explainable rules have a belief value equal to 0). This belief value will be combined with the expected accuracy (CN's usual heuristic) to provide a measure of rule quality, representing both belief-weighed explainability and accuracy. This quality measure will be used to evaluate the rules considered during search. The advantage of this approach is that both explainable and non explainable rules are treated the same way. Moreover, it is possible control how much explainable rules are preferred to non explainable ones. This trade-off between accuracy and explainability can be controlled by a simple, numerical parameter of the search. This approach will also make it possible to set preference values to links in the model: a less preferred link will have to be more strongly confirmed by training data to be used in an explanation of a learned rule.

One last question we would like to address is: where do the models come from? One could imagine if insistence on explainability prevails, developers of learning systems will then provide core initial models for some domains (e.g. finance, process control, some standard medical domains) Knowledge engineers involved in applications of ML will adapt these models to their applications, or develop their own models as they acquire the initial knowledge about the domain prior to using a learning system.

Acknowledgments

The authors are indebted to Peter Clark for his help and guidance in getting this project off the ground and establishing its goals, and for his help in the use and of the system referred here as MOBS. Rob Holte and Franz Oppacher provided useful comments about this work. The authors acknowledge the support of the Natural Sciences and Engineering Research Council of Canada.

References

[Aha Riddle 95] Aha, D., Riddle, P., Working Notes for Applying ML in Practice: A Workshop of the 12yh ICML, TR AIC 95-023, Washington, DC, Naval Research Labs, 1995.

[Bratko 95] Bratko, I., "Machine Learning: Between Accuracy and Interpretability", in [Aha and Riddle 95], p. vii, 1995.

[Bratko et al., 1989] Bratko, I., Mozetic, I., and Lavrac, N. *Kardio: A Study in Deep and Qualitative Knowledge for Expert Systems.* MIT Press, Cambridge, Ma.

[Clark and Boswell, 1991] Clark, P. and Boswell, R. Rule induction with CN2: Some recent improvements. In Kodratoff, Y, editor, *Machine Learning - EWSL-91*, pp151-163, Berlin. Springer-Verlag.

[Clark and Matwin, 1993a] Clark, P. and Matwin, S. Using qualitative models to guide inductive learning. In *Proceedings of the Tenth International Conference on Machine Learning*, pp49-56, Amherst, MA.

[Clark and Matwin, 1993b] Clark, P., Matwin, S., "Learning Domain Theories Using Abstract Background Knowledge", Machine Learning: ECML-93, P. Brazdil, ed., pp. 360-365, 1995.

[Clark and Niblett, 1989] Clark, P. and Niblett, T. The CN2 induction algorithm. *Machine Learning Journal*, 3(4), pp261-283.

[Evans and Fisher, 1993] Evans, R. R. and Fisher, D. Overcoming process delays with decision tree induction. *IEEE Expert*, 9(1), pp60-66.

[Forbus, 1984] Forbus, K. D. Qualitative process theory. *Artificial Intelligence*, 24, pp85-168.

[JKTech Ltd., 1991] *The JKSimMet Steady State Mineral Processing Simulator*. JKTech, QLD, Australia.

[deKleer 92] deKleer, J., "Qualitative Physics", in *Encyclopedia of AI*, S. Shapiro, ed., pp. 1149-1158, Wiley, 1992.

[Kuipers 95] Kuipers, B., "Qualitative Reasoning", MIT Press, 1995.

[Marchand, 1995] Marchand. Industrial Experiences in Comprehensibility. *IJCAI'95 Workshop in Machine Learning and Comprehensability*, pp125-131.

[Michalski et al. 86] Michalski, R., Mozetic, I. Hong., J., Lavrac, N., "The multi-purpose Incremental Learning System AQ15 and its testing application to three medical domains", Procs. of the *Fifth AAAI*, pp. a1041-1054, 1986.

[Ortega and Fisher, 1995] Ortega, J. and Fisher, D. Flexibly exploiting prior knowledge in empirical learning. *International Joint Conferences on Artificial Intelligence* 1995 (IJCAII).

[Quinlan, 1993] Quinlan, J. R. *C4.5: Programs for Machine Learning*. Morgan Kaufman, San Mateo, CA.

[Rouget, 1995] Rouget, T. *Learning Explainable Concepts in the Presence of a Qualitative Model*. Master's thesis, Department of Computer Science, University of Ottawa, available at http://csi.uottawa.ca/~trouget/thesis