Högdalenverket: Applying ILP in an Industrial Setting *

Robert Engels **

Dept. of Computer and Systems Sciences
Stockholm University
Electrum 230, 164 40 Kista, Sweden

Abstract. When applying Inductive Logic Programming techniques in real-world settings, many problems will come up. Since not many real-world applications are reported using ILP-techniques this report describes some problems that may appear when starting such a project. The project was done in co-operation with Högdalenverket, a heat and power plant burning household refuse in the Stockholm area, Sweden. The application problems with collecting data and the application of ILP-techniques are discussed. Results of tests performed while using SPECTRE, an ILP-algorithm developed at Stockholm University, are reported. These results show that the addition of background knowledge and addition/retraction of parameters has positive effect on the performance of the ILP-techniques. After initial tests a knowledge acquistion stage was started. This resulted in knowledge about the domain that was used for evaluating the results of learning using SPECTRE. This prevented several mistakes by interpreting data and evaluating performance. This knowledge was partly used as background knowledge for the learning algorithm. Noise handling was applied and increased the efficiency of the SPECTRE-algorithm. The paper lists the results of tests performed on data from the refuse-burning plant. It also compares the performance of the SPECTRE-algorithm on theoretical databases with the performance on this practical domain. It also gives some insight in the change of performance that can be expected when transferring from the laboratory-domains to domains gathered in the real-world.

Keywords : *Machine Learning, Inductive Logic Programming, Knowledge Acquistion*

* Published in the Proceedings of the Workshop on Intelligent Adaptive Systems, Melbourne Beach, 1995
** The author's current address is: Institute AIFB, University of Karlsruhe, D-76128 Karlsruhe, Germany.
E-mail: engels@aifb.uni-karlsruhe.de Tel: +49-721-608 4062 / Fax: +49-721-693 717

1 Introduction

In the current paper we describe the problem of applying *Machine Learning* (ML) to a real-world domain. The problem domain was provided by Stockholm Energy, a Swedish company that employs several energy-producing plants. The actual domain was provided by Högdalenverket, a refuse burning plant in the Stockholm area. The aim of the project is to find control rules for the *Selective Non Catalytic Reduction*-techniques (SNCR) used at the plant in order to lower the NO_x-emmision. *Inductive Logic Programming* (ILP) was used for learning in the domain. Few real world applications using ILP are available for evaluation. The algorithm used for this application is called SPECTRE[3] and is developed at the Stockholm University [1]. Modelling the domain using ILP-techniques only was not convenient, so it was decided to add a knowledge acquistion stage. At this stage interviews at the plant and knowledge available from reports etc. were taken into consideration. The aim of the application of the ILP-techniques and the Knowledge Acquistion stage was to try to build and refine a model of the processes at the Högdalenverket power plant. This formalization of the processes could then be used at a later stage of the project as an input-model for learning control rules for directing SNCR-processes.

2 Using ILP-techniques: the SPECTRE algorithm

The ILP-technique used was the SPECTRE algorithm [1]. This algorithm is based on the idea of finding an inductive hypothesis through the specialization of a logic program. The program has as input a logic program, a set of negative examples and a set of positive examples. Then the specialization problem (as commonly adopted in ILP) is:

Given: a definite program P. Let E^+ be a set of positive examples and E^- be a set of negative examples such that $E^+ \cap E^- = \emptyset$.

Find: a specialization of P (called P') such that $M_{P'} \subseteq M_P$ while $E^+ \subseteq M_{P'}$ and $E^- \cap M'_P = \emptyset$.[4]

The algorithm searches for a hypothesis by pruning an SLD-tree. The shape of this tree is defined by the computation rule. Boström and Idestam Almquist give a discussion on SPECTRE as well as the results of running SPECTRE on domains taken from the UCI repository of machine learning databases [1]. They test the SPECTRE algorithm using three different computational rules: the standard PROLOG computation rule (which selects the left-most literal), a computation rule that selects literals at random and a computation rule that applies unfolding and than minimizes the residual impurity of the resolvents. The unfolding and pruning performed by the last computation rule is as follows:

Given is a clause G of a definite program P. Let literal Am in clause G be the literal to unfold upon.
Then unfolding is performed by replacing clause G with the resolvents G' of G and with all clauses in P that have a head that is unifiable with Am.

By applying unfolding repeatedly it is possible to make every (sub-)tree of the SLD-tree available for pruning. The pruning is performed by cutting the appropriate tree in the SLD-tree. In practice this means that after performing unfolding, the pruning can be realized by removing the literal

[3] SPECialization by TRansformation and Elimination
[4] $M_{P'}$ denotes the least Hebrand model as described in [7]

corresponding to the sub-tree to be removed. An more extensive description of the SPECTRE algorithm is provided by Boström and Idestam-Almquist [1]. In the current paper these experiments will be extended by research involving the application of the SPECTRE algorithm on a real-world domain (See section 3). The results of these experiments are compared with the results reported in [1].

3 The industrial setting: Högdalenverket

The domain which was evaluated contains the logged data of the P3 combustion chamber at Högdalenverket, a refuse burning plant in the Stockholm area. The examples consist of instances of the target clause T. Each example has an initial 27 parameters. All of these are measurements involving the filtering and cleaning of emission gasses. The plant consists of three combustion chambers with a total heating capacity of 100 MW altogether. The maximum incineration capacity of the plants is 37 tonnes of refuse per hour. The problem is provided by the Swedish Government; laws for the maximum-level of NO_x that emision gasses from the plant may contain are going to be more restrictive in the near future. The gasses are cleaned partially by the use of several filter and injection systems. One of the processes that is involved is the *Selective Non Catalytic Reduction* process (SNCR). While performing SNCR, ammonia is injected in the combustion chamber in order to control the formation of NO_x. Injection is performed at 11 sites on 2 different levels. This is done in order to be able to inject at the place with the optimal temperature for the SNCR-processes in the combustion chamber. The amount and place of the ammonia injected influences the actual performance of the SNCR-process. Due to the unknown quality and the changing quantity of the refuse used the amount of required ammonia changes continously. The amount of ammonia which causes a lower NO_x emission depends on several parameters.

SNCR-techniques require a nitrogenous chemical to be injected in the burning process. The nitrogenous chemicals used for SNCR are mainly ammonia (NH_3) and urea (NH_2CONH_2). Ammonia is used at Högdalenverket. One of the main problems with applying SNCR techniques is the small temperature window in which these techniques perform optimally. If the temperature is too low or too high there will be unwanted side effects. These vary from a high NH_3-slip towards a higher NO_x emission, an effect which is unwanted. The main chemical reactions that take place in the SNCR-area are:

$$6NO + 4NH_3 \iff 5N_2 + 6H_2O \qquad \text{(a)}$$

$$6NO_2 + 8NH_3 \iff 7N_2 + 12H_2O \qquad \text{(b)}$$

There is always a trade-off between NH_3-slip and the reduction of the amount of NO_x in the emission gasses. This means that an increased injection of NH_3 causes an increase in performance of the SNCR-processes. However, doing so will result in a higher NH_3-slip (see reactions (a) and (b)). The higher performance when injecting more ammonia is caused by the higher availability of NH_3 for reaction with NO and NO_x. This higher availability of NH_3 also causes a higher slip while not enough NO and NO_x is available for reaction. It is clear that a way has to be found to control the amount of injected ammonia within small limits. The way Högdalenverket regulates this process presently is to couple the injection of NH_3 to one overall temperature measurement seen as representative for the temperature in the combustion chamber. That this is not an optimal solution is shown by the test-results which show the lowest accuracy when using this overall temperature measurement as the only parameter to predict from.

4 Högdalen and Inductive Logic Programming

4.1 Symbolic learning at Högdalen

There are several reasons for applying ILP-techniques at the Högdalen domain. After developing the SPECTRE algorithm it was tested on several domains that were taken from the UCI repository of machine learning databases and domain theories. During this evaluation SPECTRE performed very well. In most cases the SPECTRE-heuristic performed better than the other ILP-heuristics it was compared with while producing fewer clauses [1]. After this research an application using SPECTRE on a real-world domain was looked for in order to get more experience with using ILP-techniques in real-world settings. The Högdalen domain was selected because of the availability of relatively large amounts of data in Excel-format. The data was logged during a period of testing at the plant, which resulted in the availablility of data logged under different (standard and non-standard) circumstances. In practice it meant that during a certain period of time the technicians varied a number of (otherwise fixed) parameters important for the combustion processes.

The Högdalen problem domain is also interesting because it deals with environmental protection and is therefore seen as a problem worth solving. Högdalen-verket was convinced that there should be a way to make sufficient use of the existing plant-control computers. This possibility to experiment with the application of new symbolic techniques provided a good opportunity to get more insight in the problems involved in applicating ILP in a real-world setting. At the same time there was a hope for some new insights in the processes involved in cleaning the emission-gasses. These ideas should then ideally be used to support understanding of the underlying processes by the plant technicians.

The specification of the problem that was defined at Högdalenverket was initiated by the fact that equipment at the plant was theoretically available for reducing of the NO_x-emission where in practice no control rules to do so were available. Due to the apparent possibility of improvement by simply finding control rules for the (already available) equipment and through the availability of already logged data it was seen as an attractive possibility to evaluate the SPECTRE algorithm in this domain.

4.2 Data transformation

The fact that the data were not represented purely symbolically, but numerical was problematic in the sense that few approaches are reported trying to integrate numerically and symbolical representation for use with ILP-techniques. Many practical domains however are represented numerically. Several authors propose solutions for such problems (see for example [9], [2]). One approach that does pay attention to the problem of numerical values introduces the use of similarity measures for the selection of cut-points for numerical attributes [9]. This approach seems to be promising for domains where the training set is not sufficiently large, i.e. were the concept boundaries are not likely to be clearly defined, (as in instance spaces with few instances). [9] also concludes that when there is a higher coverage of the concept boundaries by the training set the use of these similarity based heuristic becomes less important. Our hypothesis was that at the Högdalen domain there is a good coverage of these concept boundaries. This was caused by the testing at Högdalen, where normally fixed parameters were varied during a certain test-period in order to get a better view of the emission effects (and thus the concept boundaries).

The data consists of measurements of 27 different parameters at a given time. The parameters in the logged data can be subject to noise. The parameters that cause the effects are not measured simultaneously. There is always a delay of several (not exactly known) seconds. The solution chosen for this problem was to take the files with the 5 minute averages instead of the 10-second loggings.

This approach filters out most of the delay effects. The reason for doing it this way is that it is not possible to measure the right time delays for every parameter. These averages are then translated into semantically more meaningful groups. The grouping is done using the intervals mentioned in figure 1. The data is corrected for incomplete and extreme data that were caused by measuring mistakes at the plant. The data were divided in subgroups according to standard statistical methods.

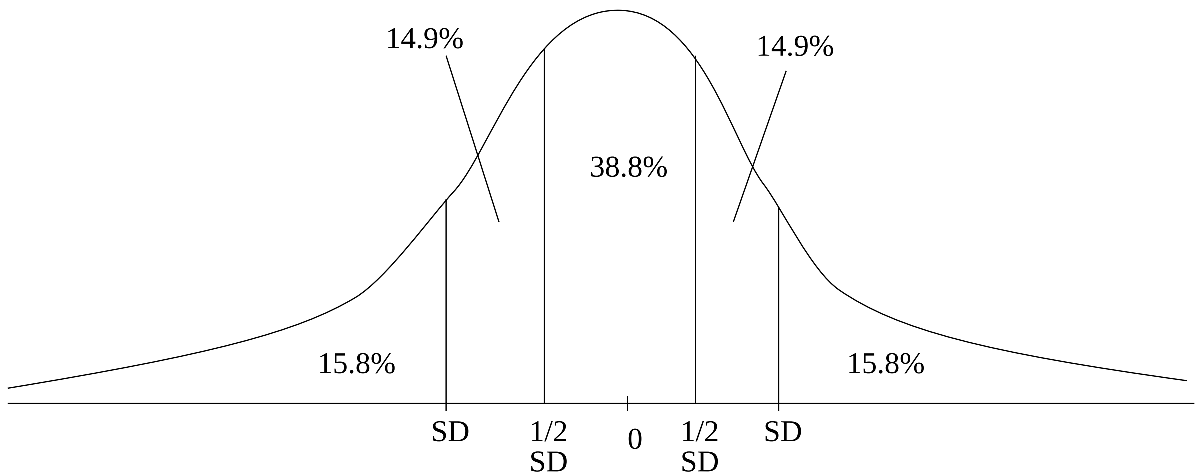

Fig. 1. Interval division based on standard deviation and normal distribution of populations.

In our approach we took the Standard-Deviation Curve and created five sectors (see Fig.1). Every numerical value fits in one sector. Each sector was given a semantic name in order to facilitate interpretation in a later stage. All literals in the examples were then processed according to this scheme.

The advantage of doing this is that the data no longer is represented numerically and that the data set became smaller. A disadvantage of doing so lies in the fact that some information is lost. On the other hand there is a lower complexity of the data set and the representation for use with a logic program is facilitated. Another advantage is that there is now a possibility to learn knowledge that is semantically useful. This effect is demonstrated at Högdalen. Grouping the data also decreases the effect that (false) extremes can carry out on the results.

4.3 ILP and Knowledge Accquisition

Most ILP-systems can be used with different quantities and qualities of background knowledge. Background knowledge includes already known relations between the domain concepts. Adding background knowledge to the specialization problem has the advantage that the search process can be biased to raise the efficiency of the search process (see [12], [8], [6], [14], [4]). When databases grow larger this topic becomes of more interest. Background knowledge can be elicited performing Knowledge Acquistion and is normally represented in a symbolical form.

The tests were initially conducted using only an overgeneral theory. Results of these learning processes without background knowledge are reported in table 1 and 2. After an evaluation the decision was made to add a knowledge acquistion stage in order to get some understanding of the domain at hand. As mentioned above adding background knowledge might increase efficiency and accuracy of the search algorithm by biased searching and avoiding (known) local maxima. Adding a knowledge acquistion stage also facilitates the process of focussing on the most informative data in the database. Focussing is found a useful process in the stage of learning from databases ([8] for more on this topic). Interviews and reports resulted in a preliminary model of the processes involved and their connections. Part of this knowledge was selected to provide knowledge implemented in the form of intermediate predicates, were another part of this knowledge was used to focus the data, i.e. delete some literals from the examples that proved to be invalid/useless. This kind of problem-solving is typical (in our view) for the application of new techniques in real-world applications. Spending some extra time on this knowledge acquistion may prove valuable in a later stage of a project.

5 Results of running SPECTRE on the Högdalen data

During a certain period of time tests with SPECTRE were run. The tests were performed in the same way as the evaluation of the SPECTRE-algorithm mentioned in [1]. Tests were performed using the standard PROLOG computational rule (i.e. selection of the left-most literal during search), a random selection of the literal and using a computational rule that minimizes residual impurity after unfolding (i.e. takes the literal where the residual impurity will be the lowest). Also reported is the effect of pruning afterwards (a usual process for minimizing overspecialisation and redundancy by removing clauses with low coverage of examples).

The tests were done with five minute averages taken from the computer-loggings. The data were transformed according to the algorithm mentioned above. The data were used as input for the SPECTRE-algorithm. SPECTRE with different heuristics was used to evaluate the performance of ILP-heuristics on the Högdalen domain. One hundred test runs were run with every heuristic. The number of examples available for testing (data equivalent to 4620 ten-second loggings) in combination with the number of literals in every example caused a complexity problem and the need for much CPU-time.

SPECTRE with:	Accuracy (%)	# clauses	Pos. acc. (%)	Neg. acc. (%)
Prolog's	58.6	507.0	77.3	31.8
Random	57.4	255.8	78.0	27.6
Entropy	60.1	66.4	70.1	46.5
Entr. with lib. prun.	59.9	19.4	62.8	56.0

Table 1. Evaluating SPECTRE with different heuristics on the Högdalen domain (Datafile=0506/0507, No. of runs=100, NO_x-level=80 mg/nm^3, neg.ex. = 41 %)

In table 1 the results of comparing the SPECTRE heuristic with several other heuristics are shown. The data were randomly divided in 50% training and 50% test data. This means that clauses with a low covering of examples are removed. The accuracy decreases slightly by doing so (0.2 % at testing), while the number of clauses decreases dramatically (from 66 to 19).

The background knowledge that was collected during the interview sessions was included in the testing. The tests evaluate the effects of the removing of literals and the addition of domain knowledge on accuracy. Redefinition of the general theory was done which included the addition of some intermediate predicates according to the model and domain theory that was gathered.

SPECTRE with: different parametersets	Accuracy (%)	# clauses	Pos. acc. (%)	Neg. acc. (%)
Without HCl + intermed. preds.	60.0	18.2	64.3	55.0
All parameters + intermed. preds.	59.7	18.0	62.9	55.6
All parameters	59.9	19.4	62.8	56.0
without HCl	61.1	19.5	63.1	58.5
Minimal set (temp only)	53.8	23.6	53.6	54.4

Table 2. Evaluating SPECTRE with different quantities of parameters and background knowledge (Datafile=0506/0507, No. of runs=100, NO_x-level=80 mg/nm^3, neg.ex. = 41 %)

SPECTRE with:	Accuracy (%)	# clauses	Pos. acc. (%)	Neg. acc. (%)
All parameters	59.9	19.4	62.8	56.0
All parameters + 20 % noise	62.3	14.6	67.7	55.0
without HCl	61.1	19.5	63.1	58.5
without HCl + 20 % noise	62.0	14.9	65.1	58.4
Minimal set (temp only)	53.8	23.6	53.6	54.4
Minimal set (temp only) + 20 % noise	53.7	19.3	56.2	50.4

Table 3. Evaluating SPECTRE without noise handling and with a 20 % threshold (Datafile=0506/0507, No. of runs=100, NO_x-level=80 mg/nm^3, neg.ex. = 41 %)

From table 2. it is clear that measuring only one overall temperature point (as done currently at the plant) is not enough to predict the amount of NH_3 that should be injected. Accuracy of running SPECTRE on the minimal set (only one temperature measurement) is on chance level. The data also shows that taking into account all the parameters does not give a better result for the prediction quality. Ignoring the HCl-measurements (after the gathering of domain knowledge it became clear that this literal could not be informative) provides us with a better performance for negative as well as positive performance. When adding the intermediate predicates (which provides extra grouping) the performance changes again. The performance is nearly as good as without adding intermediate predicates, but the learning produces less clauses (which can be seen as an improvement of performance, since few clauses is one of the goals of classification algorithms). This result makes it likely that adding more domain knowledge in this form could increase performance even more.

For the first runs with the Högdalen data, no incorrect classifications were allowed. However, since the data contains noise it is unlikely that a 100 % correct classification in this domain can be found. This is of course a characterization of many real-world domains. Therefore, tests were run with noise handling. Noise handling was performed by accepting clauses as they were covering more than 80 % of the positive examples and excluded more than 80 % of the negative examples. The results of these tests are mentioned in table 3.

It is to be expected that the results are more accurate when allowing for examples to be incorrectly classified. As seen in table 3 there is a slight improvement in accuracy, but a much greater improvement in the number of clauses. This means also that in the no-noise situation rules were created that covered only a very few examples, which is an unwanted situation (i.e. expensive in CPU-time if to be considered in a final implementation of the rules).

6 Conclusions

This paper presents research concerning the application of an Inductive Logic Programming technique in a real-world setting. The project is still under continuation. The results that are mentioned are those found after the first year of the project. The paper reports several problems and results of tests that were made during the process of trying to apply inductive techniques in a real-world (industrial) problem.

One of the reasons for applying inductive techniques for this particulair domain was that a previously reported inductive algorithm was implemented and evaluated on several theoretical domains. This algorithm was to be evaluated on data concerning real-world domains during the course of this project. Another reason for using inductive learning was that the output of the knowledge acquistion process would be in the form of models containing symbolic knowledge. The representation of learned knowledge will also be in a symbolic format. This means that the knowledge is easily interpreted by experts at the actual site. A result of such is that the testing and evaluation is less time-consuming.

Much time has been spent on trying to get the initial data. Logging the data was not a bottleneck since all the equipment was already available and operational. A problem was the incompleteness of the logged data. Much of this incompleteness was caused by broken or badly functioning equipment, false data, etc. At this point the advantage of a knowledge acquistion process became apparent. The interviews and reports made it possible to recover from many of the initial problems concerning the data. Focussing on certain parameters was one of the results of this. This re-evaluation of the data resulted in a new series of loggings at Högdalenverket. The data that was finally provided formed a sufficient basis for testing.

The use of knowledge acquistion is seen as valuable when applying ILP in real-world settings. It is seen as important to have reliable and effective methods to model domain knowledge. Modelling domain knowledge has several advantages. The knowledge provided by the experts and their (literature)sources during the knowledge acquistion stages gives a good idea of where to look for possible problems, and above all, ways to deal with them. Many of the problems that came up are absolutely not unique and the experts often have good ideas about possible points of attack. Although interviewing experts and processing reports does mean an extra time-consuming stage in a project, this can be useful and time-saving during the later stages of a project. A second advantage of doing so is that modelling background knowledge results in a better understanding of the problem at hand. In our project using background knowledge decreases the amount of clauses needed for the classication of the examples of the data-set. This is seen as an indication that the modelling and adding of more background knowledge at later stages of the project is preferable. The low performance on learning when only taking into consideration the main temperature measurement is seen as an indication for the necessity of taking into account more parameters. The results of the second test series give an indication of which parameters should be taken into account.

The results are not yet good enough to justify an implementation of the found rules. According to the view of [2] these results are to be used for evaluation and therefor to be seen as an intermediate step in development of a good ruleset. One of the questions raised at this point is whether it might be a good idea to redescribe the database representation (i.e. raising the level of representational granularity as described in [2]).

However, in this report it is shown what the drawbacks on planning can mean for the application of ILP-techniques. It is also shown what the impact is of certain manipulations of the data for the performance. After the interviews the test series without the HCl parameter were initiated. These test series showed that this parameter was unusable. The test series show an increase in performance when this parameter is not taken into account. It was very hard to get data of test runs at Högdalen without faulty measurements and therefore it became clear that noise handling would be necessary when dealing with the Högdalen domain.

There was a hope of finding results that might be transferred back to the plants technicians in the form of new, unknown concept-relation descriptions. This was based on reported successes of using inductive learning algorithms to find knowledge that is seen as new and interesting by experts and that triggered new knowledge (as reported in [5] and [10]).

Evaluating the project, it can be stated that using inductive learning algorithms on real-world domains without any knowledge acquistion process is not very helpful. As mentioned in [2] applying machine learning algorithms on "raw" data extracted from a database is of limited benefit. Our experience supports this point of view and we expect much benefit from the knowledge elicitation and representational redescription processes as described in [2].

The study as done till now is only an intervening stage of the project. The next stages of the project will include some further testing on newly provided data from Högdalenverket and testing the Högdalen domain with the MOBAL-system [11]. It is expected that performance can increase using better noise handling and more accurately measured data (which presumably becomes available soon). Representation of the learning problem as a problem of increasing/decreasing the ammount of ammonia injected at the several injection points is also planned. As mentioned before, it is also the intention to model and add more background knowledge in the course of the project. The combination of a knowledge acquistion stage and ILP has shown its value and will be needed in later stages of the project as well.

Acknowledgements

The work has been supported by the Swedish National Board for Technical and Industrial Development (NUTEK) as project "Machine Learning Techniques for Program Development" number 9303237/9405194, the Stockholm University and the University of Amsterdam. The help of Henke Boström with the application of his SPECTRE-algorithm and his comments on earlier drafts of this paper were invaluable. Thanks to Mark Tazewell for correcting my mongrel english... (That's what he calls it, and now it is Aussie-english, I guess).

References

1. Boström, H. and Idestam-Almquist, P., "Specialization of Logic Programs by Pruning SLD-Trees", *Proceedings of the 4th International Workshop on Inductive Logic Programming*, volume 237 of *GMD-Studien, Gesellschaft für Mathematik und Datenverarbeitung MBH* (1994) 31–48
2. Cupit, J. and Shadbolt, N., "Representational redescription within knowledge intensive data-mining." *Proceedings of the Third Japanese Knowledge Acquisition for Knowledge Based Systems Workshop*, Hatoyama, Japan, nov 7th – 9th, (1994)
3. Feng, C., "Inducing Temporal Fault Diagnosis Rules from a Qualitative Model" In: Muggleton, F.H. *Inductive Logic Programming* (1992)

4. King, R.D., Muggleton, S.H., Lewis, R.A. and Sternberg, M.J.E., Drug design by machine learning: The use of inductive logic programming to model the structure-activity relationships of trimethoprim analogues binding to dihydrofolate reductase. *Proceedings of the National Academy of Science USA*, volume 89, 11322–11326 (1992)
5. Lindner, G., "Logikbasiertes Lernen in relationalen Datenbanken." (in German) *LS-8 Report 12*, University Dortmund (1994)
6. Lloyd, J.W., "Foundations of Logic Programming." Springer-Verlag (1987)
7. Matheus, C.J., Chan, P.K. and Piatetsky-Shapiro, G., "Systems for Knowledge Discovery in Databases." *IEEE Transactions on Knowledge and Data Engineering*, volume 5, number 6 (1993), 903–913
8. Merckt, Th. van de, "Decision Trees in Numerical Attribute Spaces" *Proceedings of the 13th International Joint Conference on Artificial Intelligence, 28 august - 3 september*, volume 2, Chamberry; France (1993), 1016–1021
9. Muggleton, S., King, R.D. and Sternberg, M.J.E., Protein Secondary Structure Prediction using logic-based machine learing. *Protein Engineering*, 5(4), 647-657 (1992)
10. Morik, K., "Knowledgeable Learning using MOBAL- A Case Study on a Medical Domain." *Real-World Applications of Machine Learning; European Conference on Machine Learning*, Austria (1993)
11. Muggleton, S. and Feng, C., "Efficient Induction of Logic Programs." *Proceedings of the First Conference on Algorithmic Learning Theory* Japan (1990)
12. Quinlan, J., "Induction of Decision Trees." *Machine Learning*, volume 1 (1986) 81–106
13. Sternberg, M.J.E., Lewis, R.A., King, R.D. and Muggleton, S., "Modelling the Structure and Function of Enzymes by Machine Learning" *Faraday Discuss.*, volume 93 (1992) 269–280
14. Strömberg, A.M. and Karlsson, H.T., "Background Paper" presented to VärmeForsk Nordisc NO_x Seminar, Stockholm, march 17-19 (1993)