Dimensionality Reduction in ILP: A Call To Arms

Johannes Fürnkranz
Austrian Research Institute for Artificial Intelligence
Schottengasse 3, A-1010 Wien, Austria
E-mail: `juffi@ai.univie.ac.at`

Abstract

The recent uprise of *Knowledge Discovery in Databases (KDD)* has underlined the need for machine learning algorithms to be able to tackle large-scale applications that are currently beyond their scope. One way to address this problem is to use techniques for reducing the dimensionality of the learning problem by reducing the hypothesis space and/or reducing the example space. While research in machine learning has devoted considerable attention to such techniques, they have so far been neglected in ILP research. The purpose of this paper is to motivate research in this area and to present some results on windowing techniques.

1 Introduction

One of the most often heard prejudices against ILP algorithms is that they are only applicable to toy problems and will not scale up to applications of significant size. While it is our firm belief that the order of magnitude of this unspecified "significant size" is monotonicly increasing in order to keep the argument alive, it is nevertheless indisputable that hypothesis spaces in ILP are usually considerably larger than in propositional learning problems. Therefore it is quite surprising that in ILP, very little research has so far been dedicated to the development of dynamical approaches for automatically reducing the complexity of a learning problem, while in propositional learning such approaches are quite common in the form of subsampling and feature subset selection algorithms.

The motivation for using such approaches is three-fold:

Memory Limitations: Almost all learning algorithms still require to have all training examples and all background knowledge in main memory. Although memory is cheap and the capacity of the main memory of the available hardware platforms in increasing rapidly, there certainly are datasets which do not fit into main memory.

Efficiency Gain: Learning time usually increases (most often super-linearly) with the complexity of a learning problem. Reducing this complexity may be necessary to make a learning problem feasible.

Accuracy Gain: It has been observed that several approaches to dimensionality reduction (like feature subset selection and windowing) may lead to an increase in predictive accuracy. The reason is that larger hypothesis spaces are more likely to allow overfitting of the data.

The purpose of this paper is to motivate research on approaches to dynamic dimensionality reduction of ILP learning problems.

2 The Dimensionality of a Learning Problem

In propositional inductive learning there are usually two sources that influence the complexity of a learning problem:

Number of Attributes: In decision tree learning, each attribute usually corresponds to one test, while in rule learning each value of an attribute corresponds to a condition that can be added to the body of a rule.[1] The size of the hypothesis space for propositional learning problems thus depends crucially on the number of attributes.

Number of Examples: Candidate rules, conditions, or nodes in a decision tree have to be evaluated on the training examples. This evaluation is usually performed by counting the number of training examples they cover and computing some heuristic estimate from these counts. Thus, the cost for learning a theory will typically increase at least linearly with the number of training examples.

The complexity of a propositional learning problem can thus be roughly measured by multiplying the number of attributes with the number of training examples.[2] Measuring the complexity of an ILP problem is more complicated, but it

[1] This, of course, only holds for symbolic attributes. For numeric attributes, the number of possible threshold tests on that attribute is usually proportional to the number of training instances.

[2] Another factor that has significant influence on the learning complexity is of course the size of the target theory. However, this dimension is usually not known and beyond the control of the learning algorithm, so we disregard it.

is clear that it also has to depend in some way on the size of the example and hypothesis space respectively.

Unfortunately, the hypothesis space for most ILP problems is theoretically infinite. However, most practical ILP algorithms provide means for syntactically or semantically restricting the hypothesis space by various mechanisms that range from simple approaches, which allow to specify modes and types for certain predicates or to limit the clause length or variable depth, to complex description languages for explicitly modeling a hypothesis space. For an overview of such approaches see [Nédellec *et al.*, 1996]. Although in some of these approaches a calculation of the size of the defined hypothesis space is possible (as e.g. in the DLAB formalism [Dehaspe and De Raedt, 1996]), its exact size is usually unknown. Contrary to propositional learning, where the number of attributes (and maybe the average number of values for each attribute) is very indicative of the size of the hypothesis space, numerous factors influence the size of a first-order hypothesis space, including the number of literals in the background knowledge, their average arity, the average number of refinements of the refinement operator, the maximum clause length, the maximum variable depth, the size of the least Herbrand model, and many more. Identifying a small subset of easily computable measures that could be used as a standardized description of the complexity of an ILP hypothesis space would in our opinion be a rewarding topic for further research.

At first thought, one would guess that at least the problem of measuring the number of examples, i.e., the size of the example space, should not be harder for an ILP problem than it is for propositional learning problems. However, there are different views in ILP on what constitutes an example. Classical approaches, like FOIL [Quinlan and Cameron-Jones, 1995], learn a target concept from positive and negative examples, which should be entailed or not entailed by the theory for the target concept. However, recently the model-based view of the ILP learning problem, which has originally been advocated for what has been called *descriptional ILP* [De Raedt and Džeroski, 1994; Wrobel and Džeroski, 1995], has also been adapted for classification learning [De Raedt and Van Laer, 1995; Blockeel and De Raedt, 1997]. In this framework, examples are interpretations, for which the learned theory has to be true [De Raedt, 1996]. Many ILP learning problems can be formulated in both settings, which would yield different estimates, when the size of the example space is measured by merely counting the number of positive and negative examples. To further complicate things, many ILP algorithms allow to omit the specification of negative examples (relying on some sort of closed-world assumption) and/or allow intensional definitions of positive or negative examples. Thus, estimating the size of the example space is also non-trivial for ILP problems.

We can conclude that finding appropriate measures for estimating the complexity of general ILP learning problems is an interesting topic for further research. Whatever such a measure will look like, it should be clear that it must in some way depend on the size of the hypothesis space and the size of the example space. Reducing one (or both) of these factors should reduce the complexity of the learning problem. The following two sections will be concerned with techniques for achieving such a reduction.

3 Reducing the Hypothesis Space

One possibility for reducing the dimensionality of a learning problem is to reduce the size of the hypothesis space. The smaller the number of possible hypotheses the smaller will the amount of nodes be that typically have to be searched before a solution is found.

The most common technique for reducing the size of this hypothesis space is to attempt to identify relevant subsets of these attributes, a process that is commonly referred to as *feature subset selection (FSS)* [Caruana and Freitag, 1994; John *et al.*, 1994; Kohavi and Sommerfield, 1995; Pfahringer, 1995]. The special case of identifying relevant values of attributes that could be used as candidate conditions in a rule learning algorithm has also been called *literal selection* [Gamberger, 1995]. FSS algorithms attempt to dynamically identify candidate conditions that are potentially relevant for the learning problem at hand, and attempt to rule out conditions that appear to be irrelevant. Using only the potentially relevant candidates for learning will reduce the complexity of the learning problem and (what is usually the motivation for doing FSS) potentially increase the accuracy.

There are other ways of reducing the size of the hypothesis space in propositional learning algorithms, such as limiting the length of the learned rules or bounding the depth of a decision tree [Holte, 1993; Auer *et al.*, 1995]. However, these static approaches have enjoyed less popularity than the dynamic approaches for identifying relevant feature subsets.

In ILP research, the opposite is the case: Research on what has been termed *declarative bias* has flourished [Nédellec *et al.*, 1996; Cohen, 1994; Dehaspe and De Raedt, 1996; Adé *et al.*, 1995], while there are almost no approaches for dynamically reducing the size of the hypothesis space. A notable exception is [Lavrač *et al.*, 1995], where an approach for propositional literal selection [Gamberger, 1995] is used in a first-order framework by transforming the first-order problem into a propositional representation [Lavrač *et al.*, 1991]. Another simple technique for first-order literal selection is used in [Cohen, 1995b], where all relations are discarded which refer to objects that occur with a low frequency in the training set. However, both approaches seem to be limited to a subclass of ILP learning problems. A major advancement in this area might for example consist in the development of a procedure that automatically prunes unpromising branches of a refinement graph in a pre-processing step. However, we do not know of any attempt to tackle this or a similar task.

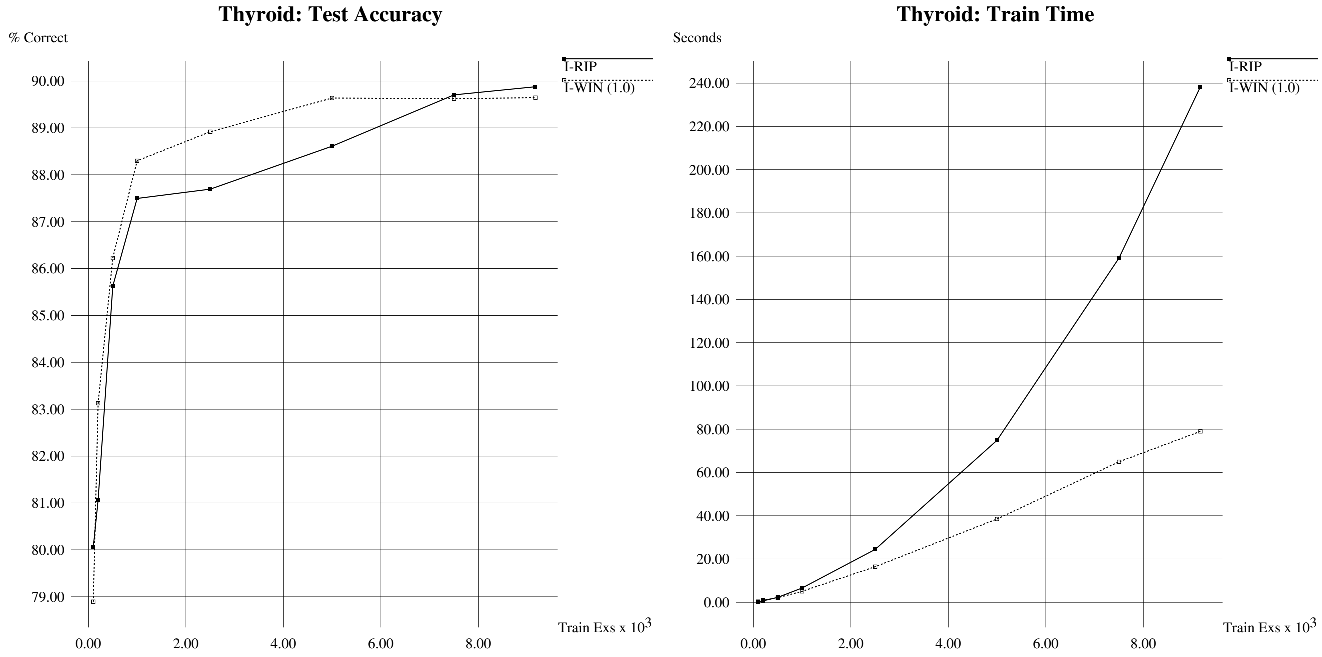

Figure 1: Results of a noise-tolerant windowing algorithm in the simplified *Thyroid* domain.

4 Reducing the Example Space

Another source of inefficiency in rule learning originates from the need to evaluate candidate rules on training examples. Another way of improving the efficiency of rule learning algorithms is therefore to use only a subsample of the available examples for learning.

Windowing is one technique for identifying an appropriate subsample to learn from. It has been proposed in [Quinlan, 1983] as a supplement to the inductive decision tree learner ID3 to enable it to tackle tasks which would otherwise have exceeded the memory capacity of the computers of those days. Despite first successful experiments in the KRKN domain [Quinlan, 1983] windowing has not played a major role in machine learning research. One reason for this is certainly the rapid development of computer hardware, which made the motivation for windowing seem less compelling. However, recent work in the areas of *Knowledge Discovery in Databases* [Kivinen and Mannila, 1994; Toivonen, 1996] and *Intelligent Information Retrieval* [Lewis and Catlett, 1994; Yang, 1996] has recognized the importance of dimensionality reduction through subsampling for reducing both, learning time and memory requirements. Other subsampling approaches include *peepholing* [Catlett, 1991], which uses dynamical subsampling at each node in a decision tree, thus extending an earlier proposal described in [Breiman *et al.*, 1984], *partitioning* [Domingos, 1996], which partitions the data into segments of equal size and combines the results obtained on each partition (similar to [Toivonen, 1996]), and *uncertainty sampling* [Lewis and Catlett, 1994], which is closely related to windowing, but does not extend the current window based on misclassifications, but on the confidence the learner has into its learned theory.

A good deal of the lack of interest in windowing can be attributed to an empirical study [Wirth and Catlett, 1988]. The authors studied windowing with ID3 in various domains and concluded that windowing cannot be recommended as a procedure for improving efficiency. The best results were achieved in noise-free domains, such as the *Mushroom* domain, where windowing was able to perform on the same level as ID3, while its performance in noisy domains was considerably worse.

Recently, we have demonstrated that rule learning algorithms are better suited for windowing than decision tree learning algorithms and have proposed improved versions of windowing for rule learning that are able to achieve significant gains in noise-free [Fürnkranz, 1997a] and noisy [Fürnkranz, 1997b] domains. The basic idea behind these approaches is to exploit the advantage of rule learning algorithms that rules are learned independently: rules that have been learned from the current window and prove to be "reliable" on the entire training set can immediately be added to the final theory. All examples covered by such a rule can be removed from the training set and the current window, thus reducing its size. This enables the windowing procedure to gain efficiency even in domains where only parts of the example space have some redundancy.

Many ILP algorithms, such as FOIL and its derivates [Quinlan and Cameron-Jones, 1995] and PROGOL [Mug-

```
procedure WINDOWING(Algorithm,LP)

RedLP = INITIALIZEREDUCTION(LP)
loop
    Theory = CALL(Algorithm,RedLP)
    Q = EVALUATE(Theory,LP)
    if STOPPINGCRITERION(Q,LP,RedLP)
        return(Theory)
    else
        RedLP = EXPANDREDUCTION(Q,LP,RedLP)
```

Figure 2: A general view of windowing.

gleton, 1995], use a separate-and-conquer learning strategy [Fürnkranz, 1997c] for accumulating the final rule set. The same learning strategy was used for the experiments described in [Fürnkranz, 1997a] and [Fürnkranz, 1997b], albeit only in a propositional setup. Figure 1 shows the results of a comparison of I-RIP, a noise-tolerant rule learning algorithm half-way between I-REP [Fürnkranz and Widmer, 1994] and RIPPER [Cohen, 1995a], and a windowed version of the algorithm in a slightly simplified, discretized version of Quinlan's thyroid domain.[3] The windowed version of the algorithm is able to outperform I-RIP in terms of both run-time and accuracy.[4] For more experimental results, including results in the pseudo-relational KRK domain, we have to refer the reader to [Fürnkranz, 1997a; 1997b].

5 A Generalized Model of Windowing

As we have outlined in the last section, we are convinced that windowing may be a powerful technique for reducing the complexity of a learning problem in domains that contain some redundancy. In the following, we will put windowing into a more general framework.

Figure 2 shows an abstraction of the principle steps of the windowing algorithm. It starts by initializing the learning problem with a reduced learning problem (e.g. with a subsample of the examples), then applies the learning algorithm to this reduced problem and analyzes the resulting theory with respect to the original problem. Unless some stopping criterion specifies that the quality of learned theory is already sufficient (e.g. if no exceptions could be found on the complete data set), the reduced learning problem will be expanded to incorporate more information (e.g. by adding all misclassified examples) and a new theory is induced. Note that this abstract framework could also be used for describing other approaches for dimensionality reduction including approaches for hypothesis space reduction. As an example think of an algorithm that attempts to learn a theory in a simple hypothesis space first and only switches to more complex hypothesis spaces if the result in the simple space in unsatisfactory. Such an approach has been realized in CLINT [De Raedt and Bruynooghe, 1990], but could also be imagined for other ILP algorithms. For example, in FOIL one could systematically vary certain parameters that influence the complexity of the hypothesis space, like the number of new variables that can be introduced in the body of a clause or the maximum length of a clause, in order to define increasingly complex hypothesis spaces. Similarly, many approaches to *constructive induction* or *predicate invention* may be viewed in this framework, if the motivation for inventing a new predicate (i.e. shifting the language bias to a more expressive hypothesis language) is the insufficiency of the current hypothesis language [Stahl, 1996]. In particular, so-called *wrapper*-approaches to constructive induction, where the theory learned in one iteration is analyzed for the construction of new features for subsequent iterations, might easily be cast into this framework [Wnek and Michalski, 1994; Pfahringer, 1994; Kramer, 1994].

With some elaboration, a general algorithm akin to the one described above could also incorporate other procedures for dimensionality reduction, like wrapper approaches to feature subset selection or the improved windowing algorithms we have described in the previous section. In particular, thinking in this framework might result in approaches that develop more general approaches to dimensionality reduction that aim at reducing both hypothesis and example space at the same time. As an example consider the *peepholing* technique introduced in [Catlett, 1991], where subsampling is used to reliably eliminate unpromising candidate conditions from the hypothesis space.

6 Conclusion

In this paper, we have tried to argue that techniques for automatically reducing the complexity of a learning problem, as they are quite common in propositional machine learning approaches, also deserve attention in ILP research. In particular, first-order equivalents to feature subset selection and windowing should be worth a deeper investigation. We have outlined several directions for future research, presented some results on windowing, and sketched of a general framework in which further research might proceed.

Acknowledgements

This research is sponsored by the Austrian *Fonds zur Förderung der Wissenschaftlichen Forschung (FWF)* under grant number P10489-MAT. Financial support for the Austrian Research Institute for Artificial Intelligence is provided by the Austrian Federal Ministry of Science and Transport.

[3] For a more detailed description of how we modified the domain see [Fürnkranz, 1997b]. The only motivation for these changes was that our implementation of the algorithms is not (yet) able to handle numerical attributes.

[4] Note that the resulting rule sets were always tested on the complete data sets, so that the accuracy estimates for the right end of the curves are resubstitution estimates.

References

[Adé *et al.*, 1995] Hilde Adé, Luc De Raedt, and Maurice Bruynooghe. Declarative bias for specific-to-general ILP systems. *Machine Learning*, 20(1-2), 1995. Special Issue on Bias Evaluation and Selection.

[Auer *et al.*, 1995] Peter Auer, Wolfgang Maass, and Robert C. Holte. Theory and applications of agnostic PAC-learning with small decision trees. In A. Prieditis and S. Russell, editors, *Proceedings of the 12th International Conference on Machine Learning (ML-95)*. Morgan Kaufmann, 1995.

[Blockeel and De Raedt, 1997] Hendrik Blockeel and Luc De Raedt. Top-down induction of logical decision trees. Technical Report CW 247, Katholieke Universiteit Leuven, Department of Computer Science, Leuven, Belgium, January 1997.

[Breiman *et al.*, 1984] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth & Brooks, Pacific Grove, CA, 1984.

[Caruana and Freitag, 1994] Rich Caruana and Dayne Freitag. Greedy attribute selection. In W.W. Cohen and H. Hirsh, editors, *Proceedings of the 11th International Conference on Machine Learning (ML-94)*, pages 28–36, New Brunswick, NJ, 1994. Morgan Kaufmann.

[Catlett, 1991] Jason Catlett. *Megainduction: Machine Learning on Very Large Databases*. PhD thesis, Basser Department of Computer Science, University of Sydney, 1991.

[Cohen, 1994] William W. Cohen. Grammatically biased learning: Learning logic programs using an explicit antecedent description language. *Artificial Intelligence*, 68(2):303–366, 1994.

[Cohen, 1995a] William W. Cohen. Fast effective rule induction. pages 115–123, Lake Tahoe, CA, 1995. Morgan Kaufmann.

[Cohen, 1995b] William W. Cohen. Learning to classify english text with ILP methods. In Luc De Raedt, editor, *Advances in Inductive Logic Programming*, volume 32 of *Frontiers in Artificial Intelligence and Applications*, pages 124–143. IOS Press, 1995.

[De Raedt and Bruynooghe, 1990] Luc De Raedt and Maurice Bruynooghe. Indirect relevance and bias in inductive concept learning. *Knowledge Acquisition*, 2:365–390, 1990.

[De Raedt and Džeroski, 1994] Luc De Raedt and Sašo Džeroski. First order jk-clausal theories are PAC-learnable. *Artificial Intelligence*, 70:375–392, 1994.

[De Raedt and Van Laer, 1995] Luc De Raedt and Wim Van Laer. Inductive constraint logic. In *Proceedings of the 5th Workshop on Algorithmic Learning Theory (ALT-95)*. Springer-Verlag, 1995.

[De Raedt, 1996] Luc De Raedt. Induction in logic. In R.S. Michalski and J. Wnek, editors, *Proceedings of the 3rd International Workshop on Multistrategy Learning (MSL-96)*, pages 29–38, Fairfax,VA, 1996. Machine Learning and Inference Laboratory, George Mason University.

[Dehaspe and De Raedt, 1996] Luc Dehaspe and Luc De Raedt. DLAB: A declarative language bias formalism. In *Proceedings of the International Symposium on Methodologies for Intelligent Systems (ISMIS-96)*, pages 613–622, 1996.

[Domingos, 1996] Pedro Domingos. Efficient specific-to-general rule induction. In E. Simoudis and J. Han, editors, *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*, pages 319–322. AAAI Press, 1996.

[Fürnkranz and Widmer, 1994] Johannes Fürnkranz and Gerhard Widmer. Incremental Reduced Error Pruning. In W. Cohen and H. Hirsh, editors, *Proceedings of the 11th International Conference on Machine Learning (ML-94)*, pages 70–77, New Brunswick, NJ, 1994. Morgan Kaufmann.

[Fürnkranz, 1997a] Johannes Fürnkranz. More efficient windowing. In *Proceedings of the 14th National Conference on Artificial Intelligence (AAAI-97)*, Providence, RI, 1997. AAAI Press. In press.

[Fürnkranz, 1997b] Johannes Fürnkranz. Noise-tolerant windowing. Technical Report OEFAI-TR-97-07, Austrian Research Institute for Artificial Intelligence, 1997. Submitted to IJCAI-97.

[Fürnkranz, 1997c] Johannes Fürnkranz. Separate-and-conquer rule learning. *Artificial Intelligence Review*, 1997. To appear.

[Gamberger, 1995] Dragan Gamberger. A minimization approach to propositional inductive learning. In N. Lavrač and S. Wrobel, editors, *Proceedings of the 8th European Conference on Machine Learning (ECML-95)*, number 912 in Lecture Notes in Artificial Intelligence, pages 151–160, Heraclion, Greece, 1995. Springer-Verlag.

[Holte, 1993] Robert C. Holte. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11:63–91, 1993.

[John *et al.*, 1994] George H. John, Ron Kohavi, and Karl Pfleger. Irrelevant features and the subset selection problem. In W.W. Cohen and H. Hirsh, editors, *Proceedings of the 11th International Conference on Machine Learning (ML-94)*, pages 121–129, New Brunswick, NJ, 1994. Morgan Kaufmann.

[Kivinen and Mannila, 1994] Jyrki Kivinen and Heikki Mannila. The power of sampling in knowledge discovery. In *Proceedings of the 13th ACM SIGACT-SIGMOD-*

SIGART Symposium on Principles of Database Systems (PODS-94), pages 77–85, 1994.

[Kohavi and Sommerfield, 1995] Ron Kohavi and Dan Sommerfield. Feature subset selection using the wrapper model: Overfitting and dynamic search space topology. In U.M. Fayyad and R. Uthurusamy, editors, *Proceedings of the 1st International Conference on Knowledge Discovery and Data Mining (KDD-95)*, pages 192–197. AAAI Press, 1995.

[Kramer, 1994] Stefan Kramer. CN2-MCI: A two-step method for constructive induction. In *Proceedings of the ML-COLT-94 Workshop on Constructive Induction and Change of Representation*, 1994.

[Lavrač *et al.*, 1991] Nada Lavrač, Sašo Džeroski, and Marko Grobelnik. Learning nonrecursive definitions of relations with LINUS. In *Proceedings of the 5th European Working Session on Learning (EWSL-91)*, pages 265–281, Porto, Portugal, 1991. Springer-Verlag.

[Lavrač *et al.*, 1995] Nada Lavrač, Dragan Gamberger, and Sašo Džeroski. An approach to dimensionality reduction in learning from deductive databases. In Luc De Raedt, editor, *Proceedings of the 5th International Workshop on Inductive Logic Programming (ILP-95)*, pages 337–354, Heverlee, Belgium, 1995. Katholieke Universiteit Leuven.

[Lewis and Catlett, 1994] David D. Lewis and Jason Catlett. Heterogeneous uncertainty sampling for supervised learning. In *Proceedings of the 11th International Conference on Machine Learning (ML-94)*. Morgan Kaufmann, 1994.

[Muggleton, 1995] Stephen H. Muggleton. Inverse entailment and Progol. *New Generation Computing*, 13(3,4):245–286, 1995. Special Issue on Inductive Logic Programming.

[Nédellec *et al.*, 1996] Claire Nédellec, Céline Rouveirol, Hilde Adé, Francesco Bergadano, and Birgit Tausend. Declarative bias in ILP. In L. De Raedt, editor, *Advances in Inductive Logic Programming*, volume 32 of *Frontiers in Artificial Intelligence and Applications*, pages 82–103. IOS Press, Amsterdam, 1996.

[Pfahringer, 1994] Bernhard Pfahringer. Controlling constructive induction in CiPF: an MDL approach. In Pavel B. Brazdil, editor, *Proceedings of the 7th European Conference on Machine Learning (ECML-94)*, Lecture Notes in Artificial Intelligence, pages 242–256, Catania, Sicily, 1994. Springer-Verlag.

[Pfahringer, 1995] Bernhard Pfahringer. Compression-based feature subset selection. In *Proceedings of the IJCAI-95 Workshop on Data Engineering for Inductive Learning*, 1995.

[Quinlan and Cameron-Jones, 1995] John Ross Quinlan and R. M. Cameron-Jones. Induction of logic programs: FOIL and related systems. *New Generation Computing*, 13(3,4):287–312, 1995. Special Issue on Inductive Logic Programming.

[Quinlan, 1983] John Ross Quinlan. Learning efficient classification procedures and their application to chess end games. In Ryszard S. Michalski, Jaime G. Carbonell, and Tom M. Mitchell, editors, *Machine Learning. An Artificial Intelligence Approach*, pages 463–482. Tioga, Palo Alto, CA, 1983.

[Stahl, 1996] Irene Stahl. Predicate invention in Inductive Logic Programming. In L. De Raedt, editor, *Advances in Inductive Logic Programming*, volume 32 of *Frontiers in Artificial Intelligence and Applications*, pages 34–47. IOS Press, 1996.

[Toivonen, 1996] Hannu Toivonen. Sampling large databases for association rules. In *Proceedings of the 22nd Conference on Very Large Data Bases (VLDB-96)*, pages 134–145, Mumbai, India, 1996.

[Wirth and Catlett, 1988] Jarryl Wirth and Jason Catlett. Experiments on the costs and benefits of windowing in ID3. In J. Laird, editor, *Proceedings of the 5th International Conference on Machine Learning (ML-88)*, pages 87–99, Ann Arbor, MI, 1988. Morgan Kaufmann.

[Wnek and Michalski, 1994] Janusz Wnek and Ryszard S. Michalski. Hypothesis-driven constructive induction in AQ17-HCI: A method and experiments. *Machine Learning*, 14(2):139–168, 1994. Special Issue on Evaluating and Changing Representation.

[Wrobel and Džeroski, 1995] Stefan Wrobel and Sašo Džeroski. The ILP description learning problem: Towards a general model-level definition of data mining in ILP. In K. Morik and J. Herrmann, editors, *Annual Workshop of the GI Special Interest Group Machine Learning (GI FG 1.1.3)*, Dortmund, Germany, 1995.

[Yang, 1996] Yiming Yang. Sampling strategies and learning efficiency in text categorization. In M. Hearst and H. Hirsh, editors, *Proceedings of the AAAI Spring Symposium on Machine Learning in Information Access*, pages 88–95. AAAI Press, 1996. Technical Report SS-96-05.