Editors' Introduction

Learning is a fascinating phenomenon of natural intelligence. In particular, humans have developed the ability to acquire knowledge, and to generalize or specify it in dependence on their actual needs. Also, humans are very good in learning their maternal languages as well as other languages. These abilities have always been considered the hallmark of intelligence. Therefore, learning is a central issue of research undertaken in the field of artificial intelligence, too. The challenge to design an "intelligent" computer has led to growing interest in learning within the computer science community. Nowadays, machine learning is sought after in a wider range of industrial and scientific applications, e.g., in molecular biology, in knowledge engineering, in financial prediction, in robotics, in pattern recognition, in natural language processing, and in machine discovery.

Various machine learning techniques have been developed during the last decade to satisfy this demand. Given this, it is important to elaborate a thorough theory to be able to provide, for example, performance guarantees. Additionally, it is necessary to develop a unified theory of learning as well as techniques to translate the resulting theory into applications. However, this is easier said than done. Algorithmic learning theory aims to develop mathematically based models of learning and to derive results within the models. Clearly, both parts are important. However, our understanding of learning is still too limited to allow the ultimate definition of learning. Therefore, the first part is of particular importance. Unlike in other areas of computer science, there are many interesting and contending models for learning giving vastly different result concerning the learnability and non-learnability, respectively, of target concepts. Nevertheless, all these models considerably help to broaden our understanding of what learning really is. As it turns out, there is at least one common feature shared by all learning models, i.e., recognizing regularities in the data provided and expressing the regularities found by compressing the data received. Usually, the data provided are incomplete. Moreover, in many applications one has to handle sometimes noisy or erroneous information.

The general theory studying learning from incomplete information is usually referred to as *inductive inference* which goes back at least to the seminal work of Solomonoff and of Gold. Inductive inference mainly emphasizes to answer what can be learned computationally at all. In particular, it studies the impact of several postulates on the behavior of learners to their learning power. The insight obtained is often valuable in enlarging our general understanding of learning. However, it is often much less satisfactory from the standpoint of computational efficiency. This led to the development of a variety of learning models emphasizing on the feasibility of learning. The present proceedings reflect a broad variety of the currently intensively studied areas in algorithmic learning theory.

Formal Language Learning

Inductive inference of formal languages is the study of algorithms that map evidence on a language into hypotheses about it. The investigation of scenarios in which the sequence of computed hypotheses stabilizes to an accurate and finite description (e.g., a grammar) of the target language has attracted considerable attention during the last decades. The source of information may vary from augmenting initial segments of the sequence of all positive and negative examples (all strings over the underlying alphabet are classified with respect to their containment in the target language), positive data only (any infinite sequence of strings exhausting just the target in the limit) to queries, i.e., the learner is allowed to ask particular types of questions to gain information concerning the target. Commonly used types of questions are membership queries (asking whether or not a particular string belongs to the target language) and equivalence queries (asking whether or not a particular finite description generates the target language and nothing else). If the proposed description does not generate the target, a counterexample drawn from the symmetric difference of the target and the proposed description is returned, too. Learning from examples usually results in identification in the limit, i.e., after having seen only finitely many examples the learner stabilizes its output to a correct description of the target. However, usually it is not decidable whether or not the learner has already converged. In contrast, learning via queries requires the learner to clearly indicate that it has learned by stopping asking questions and outputting a correct description of the target.

The invited paper by Yasubumi Sakakibara surveys grammatical inference with an emphasis on constructive approaches rather than enumerative ones (cf. Section 1). Starting with the learnability of regular languages and subsets thereof the paper introduces various models of grammatical inference including but not restricted to the classical model of learning in the limit and Angluin's minimally adequate teacher. Next, the learnability of context free languages from structural information and membership as well as equivalence queries is considered and the author gives a detailed account of his and his co-workers contributions to this interesting area. Subsequently, we are guided to the emerging and important field of inferring stochastic grammars which allows nice applications in molecular biology. Finally, the author deals with non-grammatical inference and outlines recently obtained results in the setting of cased-based learning of formal languages.

Section 2 starts with Fahmy and Ross' paper on efficient learning of real time one counter automata from equivalence and membership queries. Note that these languages are in general not regular. The resulting algorithm is more efficient than the known one for deterministic one counter automata. Applying Fahmy's decomposition theorem the learning can be divided into two steps. First, an initial segment of the Nerode-automaton is inferred by using Angluin's algorithm. Then, this segment is decomposed into the control and data structure of the target real time one counter automaton.

Koshiba, Makinen and Takada prove strongly deterministic even linear languages to be learnable from positive data. The target class of languages is a proper subset of the set of all context free languages. First, the target class is characterized in terms of a universal grammar and reversible control sets. The resulting learning algorithm is a generalization of Angluin's work on reversible automata and Takada's approach to learn even linear grammars.

Sakamoto introduces the notion of characteristic examples for parenthesis grammars, i.e., examples which exercise every production in the target grammar for their derivation. The main result is a learning algorithm that, given characteristic examples as input, learns by additionally asking membership queries. The construction resembles Sakakibara's approach to learn context free languages from structured examples by asking membership and equivalence queries. This also shows that characteristic examples are quite helpful, since they allow to avoid asking equivalence queries.

The polynomial time learnability of unions of at most k tree patterns via membership and equivalence queries is established by Arimura, Ishizaka, and Shinohara. Note that neither membership nor equivalence queries alone are sufficient to achieve learning of the same target class in polynomial time. Furthermore, the authors introduced a technique called *conservative refinement* that allows updating of the actual hypotheses in a way such that the size of the conjectured number of tree patterns in the target is never exceeded.

Further papers dealing with language learning are presented in Sections 6 and 10. Meyer studies probabilistic language learning of recursively enumerable families of uniformly recursive languages (abbr. indexed families) from positive examples. In this setting, the learner is allowed to flip a coin each time it reads a new example, and to branch its computation in dependence of the outcome of the coin flip. This model has been initiated by Freivalds in 1979. Subsequently, various authors studied the learning capabilities of probabilistic learners in dependence on the success probability required. Interestingly enough, all approaches undertaken so far led to discrete hierarchies. Looking at probabilistic learners that are required to fulfill the subset principle to avoid overgeneralization (guesses describing proper supersets of the target) Meyer discovers a *new* phenomena, i.e., the learning capabilities have a *dense* structure and are strictly decreasing as the success probability converges to 1.

Stephan deals with noisy inference from both positive and negative and only positive examples. The model of noise is as follows. Correct data are required to be presented infinitely often while incorrect data (the noise) is restricted to be presented only finitely often. The main result shows that learning in the limit with noise from positive and negative examples is equivalent to finite learning from noise-free data provided the finite learner has access to an oracle for the halting problem. Variants thereof are proved for learning from noisy positive examples.

Finally, Kobayashi and Yokomori investigate the approximate identification of languages in the limit from positive data. The starting point of their research is to take into account that prior knowledge concerning a suitable hypothesis space is often not available. Hence, one might be forced to perform learning with respect to an *a priori* chosen hypothesis space possibly not containing the target. Clearly, under such circumstances it seems highly desirable to learn the best possible approximation available. Therefore, the authors investigate the learnability of arbitrary target languages with respect to some given indexed family as hypothesis space. Different formalizations of "best possible approximation" are introduced. The resulting learning models are completely characterized with respect to the topological properties the relevant hypothesis spaces must possess in order to achieve approximate learning.

Interestingly, results obtained in language learning can be sometimes applied to solve learning problems arising in different areas. Gavalda and Guijarro (cf. Section 7) study the learnability of Boolean functions. The description chosen are ordered binary decision diagrams, a representation of Boolean functions that has nice computational properties. Assuming a fixed ordering over the set of variables, their algorithm learns a minimal ordered binary decision diagram with respect to that ordering for the target

Boolean function by asking polynomially many equivalence and membership queries. This result is obtained by a reduction to learning regular languages represented by deterministic finite automata. Moreover, since the range of the reduction does not exhaust the whole class of deterministic finite automata the number of queries needed can be reduced.

Learning logic programs

Logic programming and learning logic programs has attracted serious attention at least during the last decade, and constitutes another core area of this conference series. The basic scenario can be described as follows. The learner is generally provided background knowledge B as well as (sequences of) positive (E^+) and negative examples (E^-) (all of which can be regarded as logic programs, too). However, E^+ and E^- usually contain only ground clauses with empty body. The learning goal consists in inferring a hypothesis H (again a logic program) such that E^+ can be derived from $B \wedge H$ while $B \wedge H \wedge E^-$ is contradiction free. Again, if growing initial segments of E^+ and E^- are provided, we arrive at learning in the limit. Variations of this model include the use of membership queries to obtain E^+ and E^- and of equivalence queries (or disjointness queries) to terminate the learning process. Alternatively, E^+ and E^- may be drawn randomly with respect to some unknown probability distribution, and the learner is required to produce with high confidence a hypothesis that has small error (both measured with respect to the underlying probability distribution).

In Section 3, De Raedt and Van Lear propose a new approach to learning first order logic programs from positive and negative examples. The key idea is to view examples as interpretations which are true or false in the target theory instead of taking them as true and false ground facts, i.e., they are taken as a Herbrand model of the target theory. As a result, the role of positive and negative examples has to be changed. This approach has been successfully implemented, and the authors provide experimental data showing that their system can learn classification rules previous systems failed to handle.

Rao studies incremental learning of logic programs. In his setting, incremental means learning predicates in terms of previously learned predicates. This is a non-standard idea. Combining this idea with the known approach to treat background knowledge as function symbols evaluated in process of resolution by E-unification he arrives at a learning systems that is more powerful than previously designed ones. (cf., e.g., Yamamoto, ALT'93).

Sadohara and Haraguchi present an algorithm for analogical logic program synthesis from positive and negative examples for linear logic programs with a certain restriction. The novelty is to search the hypothesis space for a program that is similar to the target by refuting inappropriate similarities. Interestingly enough, the idea to refute hypothesis spaces born in the field of language learning (cf. Mukouchi and Arikawa, ALT'93) proved to be useful in this rather different setting.

Learning logic formulae

The polynomial time learnability of CNF and DNF formulae is a long standing open problem in algorithmic learning theory. These classes are of special interest, since they provide a natural and universal description of all Boolean concepts. In Section 3, Miyashiro *et al.* contribute to the analysis of the DNF learning problem.

They consider the learnability of orthogonal F-Horn formulae defined as follows. An F-Horn clause is a disjunction of negative literals and at most one function from the function class F. A k-F-Horn formula is the disjunction of at most k F-Horn clauses with distinct sets of literals (called body). Orthogonal F-Horn formulae have pairwise incomparable bodies with respect to set inclusion. This concept class is a natural generalization of k-quasi Horn formulae. If k-quasi Horn formulae could be proved to be exactly learnable using membership and equivalence queries than DNF formulae are PAC-learnable (cf. Angluin, Frazier and Pitt, *Machine Learning* 9, 147 - 164, 1992). In particular, Miyashiro *et al.* extend the latter result by showing the following. If F is the set of monotone ℓ-CNF formulae then the exact learnability of k-F-Horn formulae via membership and equivalence queries implies the PAC learnability of CNF and DNF with membership queries.

There is another paper closely related to the learnability of monotone DNF, i.e., learning minor closed graph classes by Domingo and Shawe-Taylor (cf. Section 7).

Inferring a DNA sequence

In his invited lecture, Ming Li addresses the problem of how to infer an original DNA sequence with high probability from erroneous copies. A satisfactory solution to this problem is of central importance to the Human Genome Project. The approach developed by Li and his co-authors Kececioglu and Tromp focuses on reconstruction of an original DNA sequence from a small number of erroneous copies, thereby dealing with insertion, deletion, and substitution errors. The main progress made is reducing the number of copies needed from exponentially many to polylogarithmically many. Furthermore, the convergence rate is proved to be polynomial in the length of the copies, and independent of the number of erroneous copies, thus considerably improving known techniques. The analysis of the algorithm developed makes heavy use of Kolmogorov complexity.

Learning recursive functions

Starting with Gold's and Putnam's pioneering work, inductive inference of recursive functions has been widely studied. The source of information provided are arbitrary sequences of input/output pairs of the target function drawn from an infinite set of recursive functions. Admissible hypothesis spaces are recursive enumerations of partial recursive functions the range of which comprises the target set, e.g., an acceptable programming system. The learning goal consists in identifying in the limit a number (program) in the relevant enumeration such that the enumerated function correctly computes the target with respect to the correctness criterion introduced. Many variations of this basic setting have been intensively studied, and the present volume provides further specifications designed to answer questions studied elsewhere. Case *et al.* look at machine induction without revolutionary paradigm shifts. In their model, a paradigm shift is associated with a significant change in the hypothesis guessed by the learner. As a first approach to model "significant" the *length* of a hypothesis is considered. Informally, a revolutionary paradigm shift is one performing an extreme variation in the size from the previously guessed hypothesis to the actual one. The influence of forbidding revolutionary paradigm shifts to the learning power is considered in two variations. First, all hypotheses generated must be not too far (measured by a recursive factor) from the minimal possible size. Second, all guessed hypotheses must be close to each other (again measured by a recursive factor). Finally, the authors combine this approach with the number of paradigm shifts expressed by the number of performed mind changes. Fixing the number of mind changes *a priory* it is shown that one more mind change can be much more liberating than removing the non-revolutionary constraint (in the second formalization).

Kalyanasundaram and Velauthapillai study an old problem that has resisted several attacks. The scenario considered deals with pluralistic learners modeled as a team. A team is said to learn successfully if one of its members infers the target function. The problem addressed is, given two teams with a bounded number of learners each of which is restricted to an *a priory* bounded number of mind changes (one bound for each team) and demanded to infer programs that are correct for all but an *a priory* bounded number of inputs (again one bound for each team), which of them, if any, can learn more than the other. This problem is usually referred to as the six parameter problem, and its complete solution is sought after for roughly 15 years. In particular, the authors provide a complete solution to the four parameter problem (both teams are requested to converge to an error free program).

The question whether or not learning devices are capable to "reflect" about their own competence is addressed by Jantke. Two formalizations are proposed, i.e., immediately reflecting and reflecting, respectively. In both cases, the learner is required to either learn the target function or to detect its incompetence in doing so. Immediately reflecting learners are demanded to detect there inability as soon as there is enough evidence that the initial segment of the target seen so far differs from all functions it can learn. Reflecting machines are allowed to use more time, and possibly more input data before detecting their failure to learn. Though seemingly contradicting our intuition, both formalizations are proved to be equivalent. Morever, the author provides evidence that reflecting about its own competence is possible if and only if there is no need to do so.

Finally, there is a paper dealing with the complexity of learning recursive functions. The model investigated goes back to Freivalds, Kinber and Smith. In their setting, a learner has access to a long term and a short term memory, respectively. Before reading the next input bit, the short term memory is cleared. All information the learner wants to maintain has to be stored in the long term memory. After having read the next input bit the learner can exploit its short term memory to perform all necessary computations to produce its guess as well as to decide which of the data it wants to store in its long term memory. While it was known that every function identifiable at all can be learned with linear long term memory and no short term memory, it remained open how much short term memory must be provided in the worst-case if the long term memory is required to be sublinear. Ambainis presents a complete solution to this problem by showing that every sublinear long term memory may force the learner to exceed any recursively bounded short term memory. As far as we are concerned this result provides evidence that a distinction between long and short term memory might be not appropriate for practical purposes.

Knowledge discovery in databases

In his invited lecture, Yves Kodratoff addresses the emerging topic of knowledge discovery in databases. Nowadays huge databases are available in various fields and there is a growing need for knowledge discovery in such huge databases. Knowledge discovery aims at transforming the information contained in the database into knowledge about it. Starting point is data mining, i.e., the discovery of patterns in the data. Once such patterns have been recognized they should be further processed with the overall goal to interpret them. Clearly, what kind of interpretation is desired may considerably vary in dependence on the envisaged users and goals. Kodratoff provides a thorough overview of the issues addressed and the techniques applied in this important field.

Miscellaneous

Auer (cf. Section 4) investigates the learnability of nested differences of intersection closed concept classes in the presence of malicious noise. The learning models dealt with are the on-line learning model (mistake bound), and the PAC model with malicious noise. The author proves matching upper and lower bounds. It is worth noticing that the tolerable noise rate exclusively depends on the complexity of the target concept *class*, and not on the target itself.

Nakamura and Miura (cf. Section 4) address the problem to learn a $GF(2)$-valued or real-valued function over a finite domain by querying the function values of the target at points the learner can choose. The complexity of learning is measured by the number of queries needed. Since the learning domain is finite, say N, every such function class can be represented by a linear space whose basis B consists of N pairwise independent functions. Then, every function from the concept class possesses a representation as a linear combination of basis functions. The size s of the function is the number of non-zero coefficients in the representation. The query complexity is analyzed in dependence on the domain size N and the target size s. Clearly, the target size depends on the basis, and so may the learnability. Lower and upper bounds are derived that are fairly close to each other.

In Section 7, Castro and Balcazar study simple PAC learning of decision lists. Simple PAC learning is the same as PAC learning except that the class of probability distributions is restricted to simple distributions, assuming that the examples are drawn with respect to the universal distribution. These distributions have been introduced by Li and Vitanyi (*Siam J. Comp.* 20, 911 - 935, 1991) who also showed that simple distributions form a wide and interesting class. In particular, the paper resolves the learnability of simple decisions list that remained open in Li and Vitanyi.

Last but not least, Pinter (cf. Section 7) resolves an open problem from Blum and Rivest (*Machine Learning: From Theory to Applications* 9 - 28, Springer 1993). The problem addressed concerns training very simple neural nets. Consider a fully connected layered neural net whose units compute linear threshold functions that consists of *one* layer and k units. Pinter shows that training such networks is already NP-complete.

Fukuoka, July 1995

Klaus P. Jantke
Takeshi Shinohara
Thomas Zeugmann