Proceedings of the Workshop "Learning 98"

Some Learning Systems are Interactive Proof Systems

José M. Sempere

Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
Valencia 46070 (Spain)

email : jsempere@dsic.upv.es

Abstract

Query learning is a method to learn concepts (i.e. grammars, formal languages) from examples and/or counterexamples provided by a teacher. Some variations of query learning methods have been presented along the time. This work is focused in what is called a Minimally Adequate Teacher (MAT), that is a system in which the teacher is able to answer two different types of queries : membership and equivalence queries. Some concept classes can be be learned by using this protocol.

By the other hand, a computational model has been presented in order to formalize the knowledge complexity of proofs : Interactive Proof Systems (IPS). There are results about the relationship between this model and some complexity classes (specially NP and PSPACE).

The relationship between IPS and MAT systems is explored in this paper. A technique to obtain IPS from MAT sistems is exposed. As a logic implication, the limited power of MAT systems is reduced to NP .

Introduction

Computational Learning Theory (COLT) is a research area in which methods to acquire knowledge automatically are explored as an attempt to substitute most of programming tasks. There have been a lot of approaches under this general framework. In this work, inductive inference [2,5,8,10] is the selected paradigm to study this general problem. Usually, under inductive inference paradigm, there are four items to be defined in order to fix a problem : a concept space, an information protocol, a succesfull criteria and a mapping to name the hypotheses about the concepts. This work is concerned with the grammatical inference approach to inductive inference. That is, the mapping to name the hypotheses and the concept space itself are defined through formal grammars and languages [9]. The information protocol is based on a teacher-learner protocol. That is, the learner can ask different types of queries to the teacher and the teacher answers the questions and (in some situations) gives additional information. The succesfull criteria is polynomial exact identification. It means that the learner can be thought as an efficient (polynomial time) process and, after some time, it guesses the correct hypothesis about the concept.

This approach to the computational learning theory can be modelized though classical oracle computations. So, under this approach, it seems that there exists strong connections between COLT and other areas (i.e. theory of complexity, cryptography, theory of recursiveness, ...).

By the other hand, in the last years, an attemp to formalize the concept of a *proof* has been made by introducing a computational model : Interactive Proof Systems, abbreviated IPS. Initial ideas can be found in [7], while good surveys about IPS can be found in [4,6].

In this work, a first approach to use IPS as learning strategies is presented. First, the MAT learning model and the deterministic IPS are presented. Later, an strategy to obtain an IPS from a MAT system is showed and, finally, some guidelines for future works to improve the initial results are exposed.

MAT learning systems

In 1987 [1], Dana Angluin presented a model to learn regular languages by using two different types of queries to a teacher. The model presented by Angluin was called a *Minimally Adequate Teacher*, abbreviated MAT, and it is formalized as follows

Definition 1. A MAT system is a learning system with two different components : the teacher and the learner. The teacher is assumed to answer two different types of queries: membership and equivalence. The learner can ask to the teacher membership queries or it can propose conjectures as equivalence queries. Both, the hypotheses and the concepts, are formal objects (i.e. formal grammars or automata). The proposed model is showed in figure 1.

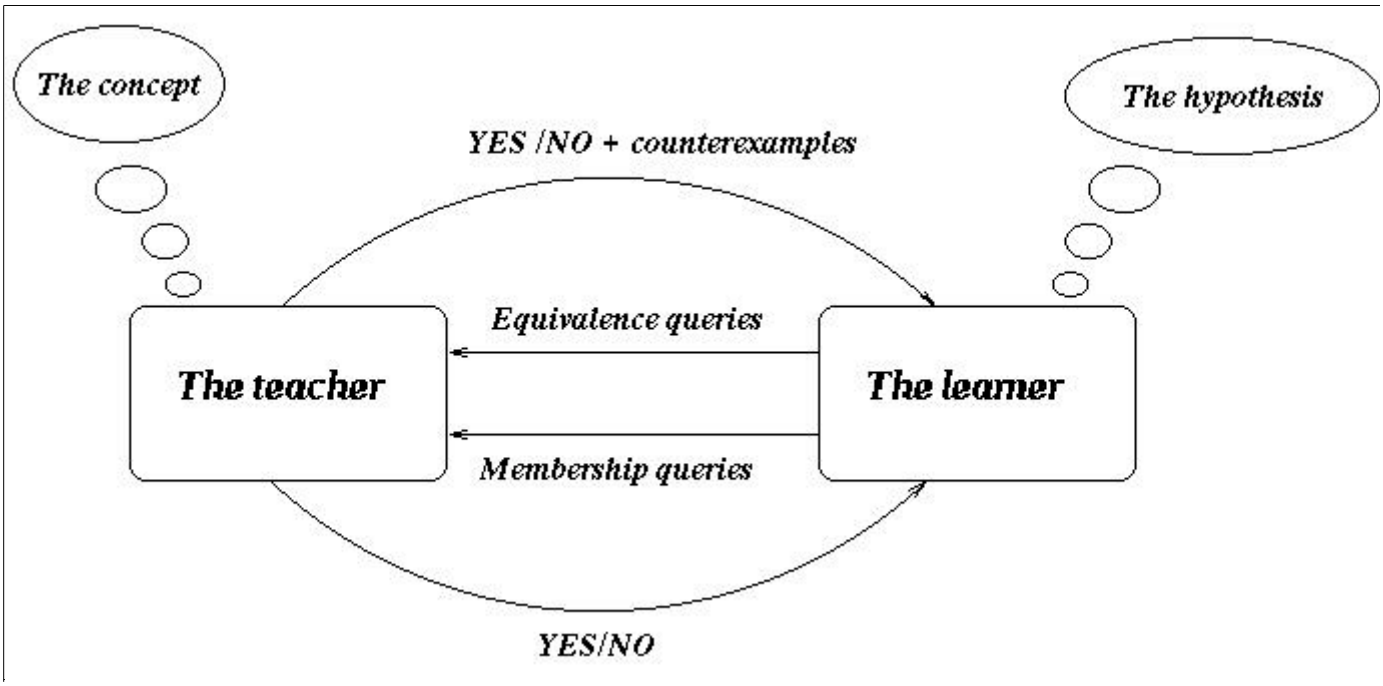

Figure 1. The MAT model.

In Angluin's work [1] it was proved that regular languages can be efficiently learned under MAT protocol.

Interactive Proof Systems

Interactive Proof Systems try to formalize the processes about how to prove any formal predicate by giving different types of knowledge to the system. It can be viewed as a two-components model (the prover and the verifier) in such a way that, given an input string x, the goal for the prover is giving enough information to the verifier to establish if the input string belongs to the language or not. The following

definition formalizes IPS

Definition 2 : An *Interactive Proof System*, abbreviated IPS, consists of two Turing machines (P and V). Every Turing machine has an input tape (this tape is shared with the other machine), a private working tape and two messages tapes, one of them for *read-only* messages and the other for *write-only* ones. P, the *prover*, is computationally unlimited while V, the *verifier*, is a polynomial time machine. The following protocol shows how a computation is performed in this model :

1 The input string is located in the input tape.
2 Both machines, P and V, take turns to perform their actions.
3 An action consists of reading a symbol of the input string, reading the message in the *read-only* tape (if there is any), writing a message in the write-only tape and perform the action as in a basic Turing machine (i.e. it changes the internal state of the finite control, it moves the tape heads and it writes a symbol in the working tape).
4 V starts to take turn for action performing.

The acceptance criteria of the model is guided by the verifier. So, when the verifier reaches a final state or simply stops its movements the computation is finished. The IPS model is showed in figure 2.

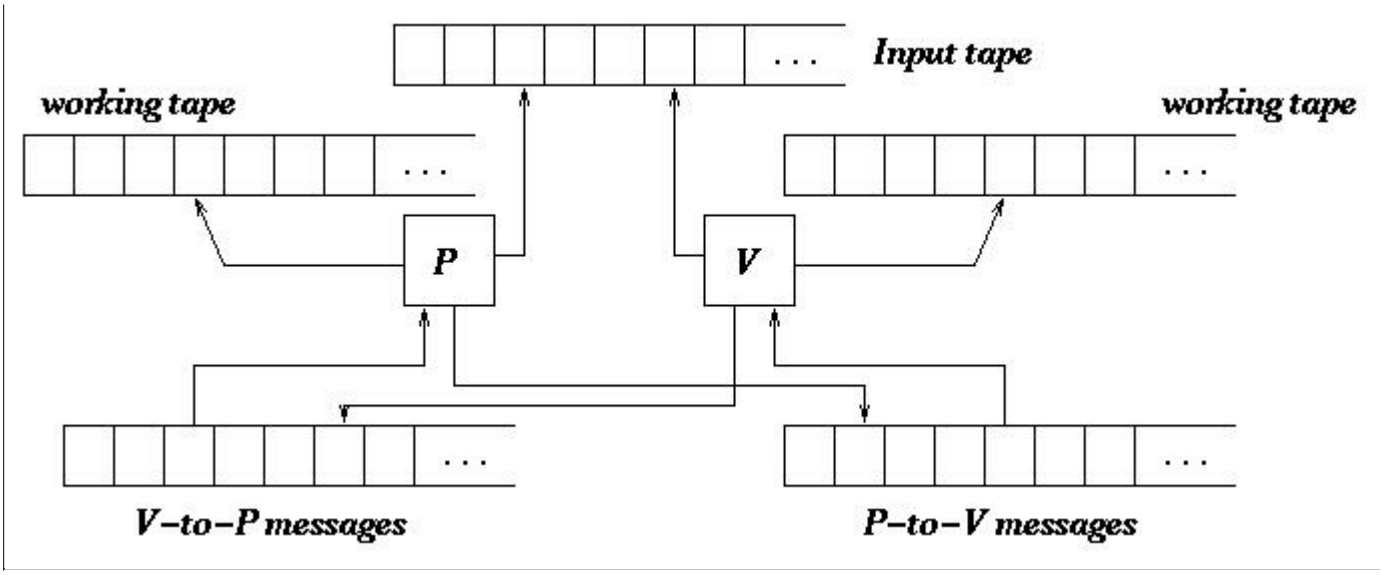

Figure 2. The IPS model.

If the *prover* and the *verifier* are both deterministic, then the IPS is a deterministic one, abbreviated DIPS. It has been proved that the class of languages accepted by DIPS is exactly *NP* .
The deterministic model can be modified by introducing randomness. If the *prover* is a probabilistic Turing machine then the obtained model is simply an IPS. It has been proved that the class of languages accepted by IPS is exactly *PSPACE* .

Proving and learning

Once the two different models to learn and prove have been presented, the relationship between them is explored. In the first place, it will be showed how to obtain a proof system from a MAT system. It will be proved that any language which can be learned under the MAT protocol can be proved by a deterministic IPS.

Theorem : Let *L* be a language which can be polynomially identified from queries and counterexamples with a MAT protocol . Then *L* can be proved by a deterministic IPS.

Proof

Let $x_1,x_2, ..., x_n$ be the membership queries that the learner asks to the teacher and let $h_1,h_2, ... h_m$ the equivalence queries (observe that h_m is the target concept, so L=L(h_m)). A deterministic IPS can be build to prove any string in L. Let us suppose that x is the input string for the IPS. The machine can be build in the sense that it simulates the actions performed by the MAT system to learn L. So, *V-to-P* messages are the membership $x_1,x_2, ..., x_n$ and equivalence $h_1,h_2, ... h_m$ and *P-to-V* messages are the teacher's answers. The order in which the messages are sent is the same in which the queries are performed in the MAT system. After some time, the verifier sends h_m to the Prover and it answer YES (it can be coded as *1*). At this moment, the only action that the verifier must carry out is just testing x in h_m . The acceptance or rejection depends on the input string and the target h_m .

So L can be accepted (proved) by a deterministic IPS.

a

A consequence of the last theorem is that MAT model can only learn those languages included in NP. It is obvious given that, if the MAT system acts in polynomial time then it tests any string in polynomial time too, so the target languages must be in P which is included in NP.
In order to learn a large class the learning protocol must be improved. Here, some guidelines to take advantage of IPS to learn new classes is exposed.

Let us suppose that I is a determinisic IPS for the language L. Let $v_1,v_2, ... v_m$ be the verifier messages and $p_1,p_2, ... p_m$ be the prover messages for an input string x. The concept to be learned is the strategy which the verifier applies to prove x. In this situation messages play an important role for the query system. In order to learn the concept L, the learner needs to ask some queries which consists of the messages $v_1,v_2, ... v_m$ together with the string x. So, the learner asks for tuples $< x, v_1,v_2, ... v_m >$ and the teacher must provide answers as $<yes, p_1,p_2, ... p_m >$ or $< no, v'_1,v'_2, ... v'_m >$. This strategy can be performed in an incremental way in the sense that the learner can ask for message v'_i and the teacher's answer should be p_i. There are two important aspects to be defined

- The encoding function for messages.
- The associated language to pairs $<v_i,p_i>$. Is it any kind of transduction ?.

Conclussions and future work

Some relationships about query learning systems and IPS have been explored. This is a first approach to establish strong connections between learning and proving. The general belief underlaying this work is that learning a concept should be equivalent to proving it. So, the future work to carry out is formalize this idea.
Furthermore, there is another concept which has not been used in this work : *randomness.* Here, the learning system should work with a probabilistic teacher. These ideas will be explored too in the near future.

References

[1] Angluin, D. *Learning Regular Sets from Queries and Counterexamples.* Information and Computation 75, pp 87-106. 1987.
[2] Angluin, D. and Smith, C. *Inductive Inference : Theory and Methods.* Computing Surveys, vol. 15. No. 3, pp 237-269. 1983.
[4] Bovet, D. and Crescenzi, P. *Introduction to the Theory of Complexity.* Ed. Prentice Hall. 1994.
[5] Gold, M. *Language Identification in the Limit.* Information and Control 10, pp 447-474. 1967.
[6] Goldreich, O. *Probabilistic Proof Systems - A survey.* 14th Annual Symposium on Theoretical

Aspects of Computer Science (STACS 97). Proceedings edited by R. Reischuk and M. Morvan in LNCS vol. 1200. pp 595-611. Springer-Verlag. 1997.
[7] Goldwasser, S., Micali, S. and Rackoff, C. *The knowledge complexity of Interactive Proof Systems.* SIAM J. Comput. Vol. 18, No. 1, pp 186-208. 1989.
[8] Osherson, D., Stob, M. and Weinstein, S. *Systems That Learn.* The MIT Press. 1986.
[9] Sakakibara, Y. *Recent advances of grammatical inference.* Theoretical Computer Science 185, pp 15-45. 1997.
[10] Solomonoff, R.J. *A Formal Theory of Inductive Inference (Part I and II).* Information and Control 7, pp 1-22, pp 224-254. 1964.