Analyzing Relational Learning in the Phase Transition Framework

Attilio Giordana ATTILIO.GIORDANA@UNIPMN.IT
Lorenza Saitta LORENZA.SAITTA@DI.UNITO.IT
DISTA, Università del Piemonte Orientale, Alessandria, Italy

Michele Sebag MICHELE.SEBAG@POLYTECHNIQUE.FR
LMS, École Polytechnique, Palaiseau, France

Marco Botta MARCO.BOTTA@DI.UNITO.IT
Dipartimento di Informatica, Università di Torino, Italy

Abstract

A key step of relational learning is testing whether a candidate hypothesis covers a given example. The covering test is equivalent to a Constraint Satisfaction Problem (CSP), which shows a phase transition in correspondence of critical values of some order parameters.

This paper investigates the effects of the phase transition in the covering test on the complexity and feasibility of learning in first order logic languages. Several hundreds of artificial learning problems have been generated. FOIL and other learners have been applied to these problems. The experiments show the presence of a failure region, where all considered learners systematically fail to identify the target concept. Furthermore, the phase transition region behaves as an attractor for the learning search, whatever the target concept and the search strategy be. Interpretations of these findings are proposed and discussed.

1. Introduction

This paper is concerned with supervised learning from structured examples, termed *relational learning* (Quinlan, 1990) or *Inductive Logic Programming* (ILP) (Muggleton & De Raedt, 1994). In relational learning the covering test — testing whether a candidate hypothesis covers a given example — can be formulated as a Constraint Satisfaction problem (CSP), which is a NP-hard task.

As a learner may generate even thousands of hypotheses, each one to be tested against every learning example, the complexity of the matching step may severely question the scalability of this type of learning. On the other hand, problem instances in a NP-hard class are not equally hard to solve. Recent work on combinatorial search shows that the hardest problem instances are concentrated in a very narrow region where a *Phase Transition* (PT) occurs (Hogg, Huberman, & Williams, 1996). The PT separates the problem space into two regions: the YES-region, where most problem instances have many solutions (and hence it is easy to find one), and the NO-region, where most problems have no solutions (and hence it is easy to prove that they are unsolvable). In correspondence of the PT the probability that a solution exists abruptly drops from almost 1 to almost 0, and the complexity of the search dramatically increases.

These findings suggested a new paradigm for the analysis of combinatorial algorithms: the focus is shifted from their worst-case complexity toward their actual complexity on the hardest problem instances, that is, those located in the phase transition region (*mushy* region).

In a previous paper, Giordana and Saitta (2000) focused on the matching step per se. They showed that a phase transition also occurs in real-world learning problems, such as the "Mutagenesis" (King, Srinivasan, & Stenberg, 1995), and they identified its location with respect to relevant learning parameters. Moreover, a comparison between the complexity of deterministic and stochastic search on problems located on the phase transition has been reported by Botta, Giordana, and Saitta (1999). In those papers, only the computational effects on learning, due to the emergence of a phase transition, have been considered. On the contrary, the goal of this paper is to experimentally investigate whether the presence of a phase transition also affects other aspects of learning, such as quality and/or performance of learned theories. The experiments have been performed by systematically sampling the space of learning problems using FOIL (Quinlan, 1990), a top-down greedy searcher with limited backtracking, to solve them. Moreover, complimentary experiments on a smaller set of problems have been done using other two learners: SMART+ (Botta & Giordana, 1993), which relies on a top-down beam-search strategy, and G-Net (Anglano, Giordana, Lobello, & Saitta, 1998), an evolutionary learner based on genetic search. Two are the main results of this paper. First, the phase transition region acts as an attractor for the search, i.e., any learner tends to explore hypotheses that ly inside the mushy region w.r.t. the training examples. Second, independently of the learner, a "failure region" appeared, i.e., a blind spot located around the phase transition, in which no learning problem could be solved. Thse results are discussed in the second part of the paper.

2. Phase Transition and FOL Learning

In this section we briefly recall previous results related to phase transition occurrence in the simplest relational learning setting (De Raedt, 1997). Let α_i, x_j, v_k denote predicate symbols, variables and constants of the application domain, respectively. We consider concept definitions restricted to conjunctions of binary literals of the form:

$$\mathcal{C} =_{def} \alpha_1(x_{i_1}, x_{j_1}) \wedge \cdots \wedge \alpha_m(x_{i_m}, x_{j_m}) \qquad (1)$$

Any learning example E is represented as a conjunction of ground literals $\alpha_h(v_{i_h}, v_{j_h})$. We say that $\mathcal{C}$ covers E iff E contains a model of $\mathcal{C}$. We will indicate $(\mathcal{C} \prec ?\, E)$ the process of testing if $\mathcal{C}$ covers E. A pair $(\mathcal{C}, E)$ is a generic instance of covering test.

Using the CSP terminology, the set of literals built on a same predicate symbol α_i in E is termed a *relation*. The complexity of example E is characterized by the number L of constants and the average size N of the relations occurring in E. Symmetrically, the complexity of any concept/hypothesis is characterized by its number n of variables and its number m of literals.

Being an ensemble phenomenon, the emergence of a phase transition depends upon the probability distribution of problem instances in the selected class. The generation of problem instances we used in this paper follows the stochastic procedure described by Botta et al. (1999). Given a set $\mathbf{A}$ of predicate names, a set $\mathbf{X}$ of variables, and a set Λ of constants, a concept description $\mathcal{C}$ can be roughly considered as uniformly extracted from the set of all formulas with the structure (1), given $\mathbf{A}$, $\mathbf{X}$, the number n of variables, and the number m of literals. Each binary relation α_i in E, corresponding to the ground instances of predicate α_i, contains N pairs of constants, uniformly extracted from $\Lambda \times \Lambda$.

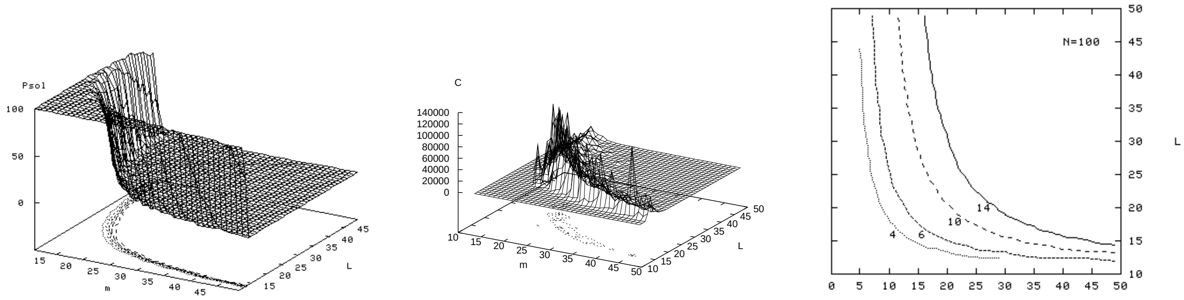
By generating a large set of covering tests $(\mathcal{C}, E)$, Botta et al. (1999) reported the appearence of a phase transition, described in Figure 1. The order parameters are the number m of literals in $\mathcal{C}$, and the number L of constants in E, whereas the number n of variables and the cardinality N of the relations have been used to parametrize the problems. Then, a covering test $(\mathcal{C}, E)$ is visualized as a point in the (m, L) plane. As it appears from Figure 1(a), the probability P_{cov} that any covering test, generated in the way mentioned above, is solvable drops from almost 1 (YES region) to almost 0 (NO region) in a very narrow region (the mushy region visible in the projection on the (m, L) plane in Figure 1(a)). In other words, any covering test $(\mathcal{C}, E)$ such that the corresponding pair (m, L) falls to the left of the PT is almost surely bound to be solvable, whereas the opposite happens for covering tests located to the right of the PT. Ideally, the "phase transition" corresponds to the contour level $P_{cov} = 0.5$. Of course, unsolvable problems may exist in the YES region, as well as solvable problems may exist in the NO region, even if very rare in a world that follows a uniform distribution.

3. Experimental Setting

The main goal of our experimention is to investigate the effects (if any) of the presence of a phase transition on learning, beyond the obvious computational increase.

The problems. A learning problem $\pi = (\mathcal{C}, \mathcal{E}_L, \mathcal{E}_T)$ consists of a target concept $\mathcal{C}$ and two sets $\mathcal{E}_L$ and $\mathcal{E}_T$ of training and test examples, respectively. As the values L and N are the same in all the examples, any covering test $(\mathcal{C}, E)$ defined by the problem π corresponds to a same point (m_π, L_π) in the plane (m, L). During learning, many covering tests are to be executes, namely one for each pair (h, E), where h is a hypothesis generated by the learner and E belongs to $\mathcal{E}_L$.

A set of 451 artificial learning problems have been constructed. In order to keep the computational cost acceptable, n is set to 4 and N is set to 100. The number m of literals in $\mathcal{C}$ ranges over $[5 \div 30]$ and the number L of constants in any example ranges over $[11 \div 40]$. The set of learning problems is thus widely spread over the



(a) (b) (c)

Figure 1. Phase transition emerging in the covering test class. (a) Plot of the probability P_{cov} that a random problem $(\mathcal{C}, E)$ is solvable vs. the number m of literal in $\mathcal{C}$ and the number L of constants in E. The graph corresponds to $n = 14$ and $N = 100$. (b) Corresponding computational complexity of the search, vs. m and L. The complexity is evaluated by the number of variable-constant unifications attempted during the search for the first model. (c) Phase transition plots ($P_{cov} = 0.5$), parametrized by the number n of variables in $\mathcal{C}$. The value N is set to 100.

YES, the mushy and the NO regions (Figure 1).

A learning problem $\pi = (\mathcal{C}, \mathcal{E}_L, \mathcal{E}_T)$ is generated as follows:

- Choose a point (m_π, L_π).
- Stochastically generate a target concept $\mathcal{C}$ with m_π literals and $n = 4$ variables.
- Stochastically generate $\mathcal{E}_L$ and $\mathcal{E}_T$, each one with 200 examples: each example contains m_π relations, each one with N pairs of constants.
- Modify part of the examples, in order to obtain 100 positive and 100 negative ones, both in $\mathcal{E}_L$ and in $\mathcal{E}_T$.

The last step of the above procedure is necessary because the generation process does not garantee a balanced example set; more specifically, when (m_π, L_π) is in the YES region, the 200 examples all come out as positive (they almost surely satisfy $\mathcal{C}$), whereas they come out as negative when (m_π, L_π) is in the NO region.

More details on the generation of the data can be found in (Botta et al., 1999).

The learners: FOIL, SMART+, G-Net.
FOIL performs a top-down exploration; it iteratively specializes its current hypothesis $\mathcal{C}_t$ by conjunction with the "best" literal $\alpha_i(x_j, x_k)$ according to some statistical criterion (Information Gain (Quinlan, 1990) or Minimum Description Length (MDL) (Rissanen, 1978)). SMART+ is also a top-down learner (Botta & Giordana, 1993); unlike to FOIL it uses a beam search with the beam width controlled by the user. G-Net is based on genetic algorithms (Anglano et al., 1998). It starts from an initial population of candidate hypotheses with different numbers of literals. As the evolutionary search of G-Net is slower than that of the other two learners, only a few experiments have therefore been performed with it [1].

Evaluation criteria.
Let $\hat{\mathcal{C}}$ denote the theory (set of hypotheses) discovered by a learner. A first issue concerns the predictive accuracy of $\hat{\mathcal{C}}$, measured by the percentage of test examples correctly classified. The predictive accuracy is considered satisfactory iff it is greater than 80% (the point of this threshold value will be discussed later on), on the test set [2]. However, one might obtain a good predictive accuracy with a theory $\hat{\mathcal{C}}$ significantly different from the true target concept $\mathcal{C}$. The second issue thus concerns the identification of $\mathcal{C}$, which is considered

[1] Another ILP learner, PROGOL (Muggleton, 1995), has also been run on some learning problems. PROGOL appeared ill-suited for large-sized artificial problems where any background knowledge was absent. We also considered using STILL (Sebag & Rouveirol, 1997), which is a bottom up learner based on the stochastic sampling of the models search space. However, it appeared that the stochastic heuristics embedded in STILL are geared toward non-uniform distributions of the examples. Further research will examine how to reshape and parametrize these heuristics depending on the distribution of the examples.

[2] The predictive accuracy was not evaluated according to the usual cross-validation procedure for two reasons. First of all, the training and test sets are drawn from the same distribution; it is thus equivalent to doubling the experiments and taking the average result, or performing a twofold crossvalidation (Dietterich, 1998). We did not double the experiments because of the huge total computational cost. Furthermore, though the result for (m, L) is based on a single trial, it might be considered significant as all other trials in the same area give the same result.

satisfactory iff the structure of $\hat{\mathcal{C}}$ is close to that of the $\mathcal{C}$, i.e., if $\hat{\mathcal{C}}$ is conjunctive.

4. Results

Predictive Accuracy. As mentioned earlier, each learning problem corresponds to a point in the plane (m, L). In Figure 2 the map of FOIL's successes and failures is reported; a cross (dot) denotes a problem $(\mathcal{C}, \mathcal{E}_L, \mathcal{E}_T)$ that FOIL successfully solved (did not solve), according to the accuracy criterion stated in the previous section. Detailed results (Table 1) show that the predictive accuracy is either very high ($\geq 95\%$) or comparable to that of random guess ($\leq 58\%$). Hence, the shape of FOIL's failure region does not critically depend on the threshold value of 80%. SMART+ has been run on about the 10FOIL's failure region, and showed a full agreement with FOIL [3]. Complementary experiments (not reported here) show that G-Net, too, fails on the same learning problems. In a nutshell, the failure region seems to be almost independent from the success criterion and the learning strategy.

These experiments suggest that relational learning succeeds iff either the target concept is sufficiently small ($m \leq 6$), or the learning problem is sufficiently far away from the phase transition. The latter condition was unexpected, as it states that for a given L, longer concepts (extreme right region) might be easier to learn than smaller ones (close to the phase transition). This point will be discussed later in Section 5.

As $\mathcal{E}_L$ and $\mathcal{E}_T$ are fixed, for the sake of simplicity we will say that a concept $\mathcal{C}$ or an hypothesis $\mathcal{C}_t$ belongs to the YES (resp, NO or PT) region when the corresponding set of covering tests $\{(C, E) | E \in \mathcal{E}_L \cup \mathcal{E}_T\}$ falls in that region.

Concept Identification. Let us look more closely at what is learned by FOIL. Table 1 first reports the characteristics of the learning problems (m and L), then the number K of disjuncts learned by FOIL, and the average number of literals in these disjuncts, denoted by $\hat{m}$. Next columns give the predictive accuracy of the learned theory $\hat{\mathcal{C}}$ on the training and test sets, and the learning run time (in seconds on a Sparc Enterprise 450). A categorization of the learning problems, explained below, is proposed in the last column.

Table 1 shows three categories of relational learning problems.

E. *Easy* problems. FOIL correctly identifies the true target concept $\mathcal{C}$ or a clause slightly more general (by at most one literal); almost all training and test examples are correctly classified. Easy problems mostly lie in the *YES* region; they might also belong to the PT region for low values of m.

M. *Manageable* problems. FOIL finds a conjunctive hypothesis $\hat{\mathcal{C}}$, which correctly classifies (almost) all training and test examples, but largely overgeneralizes $\mathcal{C}$ (e.g., $\hat{\mathcal{C}}$ has 6 literals instead of 18). Manageable problems are mostly in the *NO* region, far away from the phase transition.

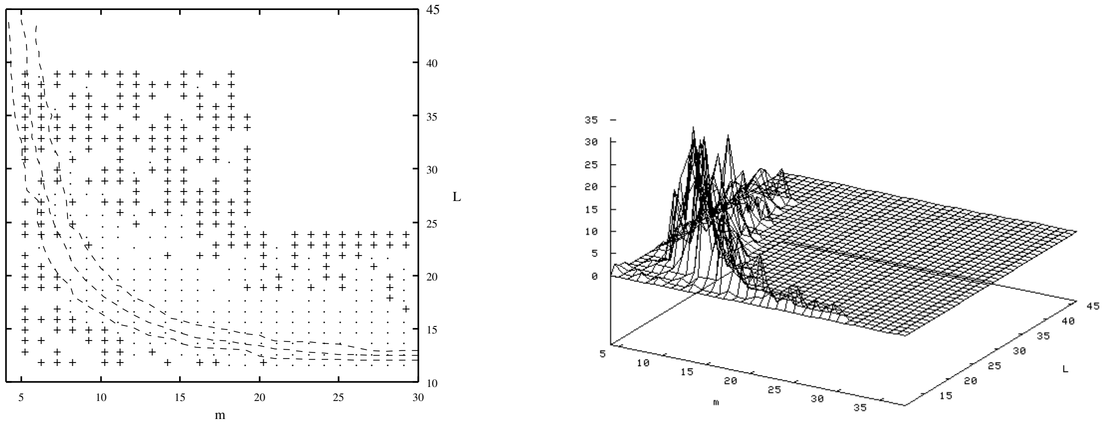
H. *Hard* problems. FOIL learns a disjunctive hypothesis $\hat{\mathcal{C}}$ involving many conjunctive hypotheses $\mathcal{C}_i$ (between 6 and 15) of various sizes, and each $\mathcal{C}_t$ only covers a few training examples. The predictive accuracy of $\hat{\mathcal{C}}$ is not much better than a random guess on the test set. The learning cost reaches its maximum for hard problems, because of the number of hypotheses learned, and of their closeness to the phase transition (see next paragraph). Hard problems fall within or close to the phase transition, for high values of m.

These results confirm the fact that a high predictive accuracy does not imply that the true concept $\mathcal{C}$ has been discovered. Obviously, there is no way one can distinguish between easy and manageable problems in real-world applications.

Table 1. Target concept $\mathcal{C}$ and learned concept $\hat{\mathcal{C}}$

$\mathcal{C}$		$\hat{\mathcal{C}}$		Performances			
m	L	K	$\hat{m}$	*train.*	*test*	CPU	
8	16	1	8	100	100	106.2	**E**
10	13	1	14	100	99	144.2	**E**
10	16	8	11.75	88	48.5	783.5	**H**
11	13	1	11	100	100	92.2	**E**
11	15	6	13.5	85	53.5	986.2	**H**
12	13	3	14	98.5	83	516.4	**M**
$\mathcal{C}$ belongs to the *YES* region (lower left)							
15	29	1	6	100	100	185.3	**M**
15	35	2	6	97.5	84.5	894.6	**M**
18	35	1	6	100	100	201.0	**M**
21	18	8	4.13	81.5	58	1394.9	**H**
25	24	1	6	100	99	135.9	**M**
29	17	1	12	100	99.5	144.9	**M**
$\mathcal{C}$ belongs to the *NO* region (upper right)							
6	28	12	8.08	91.5	50.5	815.4	**H**
7	28	11	7.63	91.5	60.5	1034.2	**H**
8	27	1	7	100	100	58.8	**E**
13	26	1	9	100	99	476.8	**M**
17	14	8	15	93	46	294.6	**H**
18	16	8	8.87	91	58.5	404.0	**H**
26	12	3	24.33	80	58	361.4	**H**
$\mathcal{C}$ belongs to the phase transition region							

[3]Unless the beam width of SMART+ is close to the size of the target concept, which means that an exhaustive search is performed.



(a) (b)

Figure 2. Results obtained by FOIL: (a) Competence Map: Failure cases (.) and Success cases (+). (b) Distribution of the location of the retained hypotheses.

Location of the hypotheses.
Figure 2(b) shows where the conjunctive hypotheses learned by FOIL belong. In the easy problems, FOIL discovers the true concept, which lies in the YES region most of the times (this is not much visible, as there are few easy problems). In the manageable problems, FOIL grasps an over-generalization of the true concept, that lies in the PT region. In the hard problems, FOIL retains seemingly random disjuncts, most of them lying in the PT. In the two latter cases, the phase transition behaves as an *attractor* of the learning search.

5. Interpretation

The above results raise at least three questions. Why does the learning search end up in the mushy region ? When and why is the target concept correctly identified ? When and why should a relational learner fail to approximate the target concept ? Some tentative answers are proposed in this section.

5.1 The phase transition is an attractor

Given a problem π, FOIL constructs a series of candidate hypotheses. Each hypothesis $\mathcal{C}_t$, coupled with every $E \in \mathcal{E}_L$, generates a covering test $(\mathcal{C}_t, E)$ located on the horizontal line $L = L_\pi$, becuase L only depends on the examples. Then, the learning process describes a path in the (m, L) plane, which, in this case, is a horizontal line $L = L_\pi$. FOIL starts with a single literal $\mathcal{C}_1$, and specializes $\mathcal{C}_t$ to obtain $\mathcal{C}_{t+1}$. The series of hypotheses thus forcedly starts in the *YES* region, then it might come to visit the mushy region, and possibly thereafter the *NO* region. Each $\mathcal{C}_t$ is required to be representative, covering sufficiently many positive examples; the last hypothesis $\mathcal{C}_T$ is such that it is sufficiently correct, covering no or few negative examples. We examine the implications of this search strategy, depending on the location of the target concept $\mathcal{C}$.

Case 1: $\mathcal{C}$ belongs to the phase transition region.
By construction, $\mathcal{C}$ would cover a random example with probability around .5; examples need little repairing (Section 3) in order to get evenly distributed training and test sets. Hence:
• No hypothesis in the *YES* region can be correct as it likely covers all training examples. The search must go on until reaching the mushy region.
• Symmetrically, any hypothesis in the *NO* region would hardly cover any training example, hence it is not representative. The search thus should stop at the very beginning of the *NO* region, and preferably before, that is, in the mushy region. Therefore, a top-down learner is bound to produce hypotheses $\mathcal{C}_T$ lying in the mushy region.

Case 2: $\mathcal{C}$ belongs to the *NO* region.
Here, negative examples do not need to be repaired; hence, any hypothesis in the *YES* region will cover them; thus the search must go on at least until reaching the mushy region. On the other hand, any hypothesis

in the *NO* region should be correct, and there is no need to continue the search. Top-down learning is thus bound to produce hypotheses $\mathcal{C}_T$ lying in the mushy region, or on the verge of the *NO* region.

Case 3: $\mathcal{C}$ belongs to the *YES* region.
The situation is different here, since there exist correct hypotheses in the *YES* region, namely the target concept itself, and possibly many specializations thereof. Should these hypotheses be discovered (the chances for such a discovery are discussed in the next subsection), it would not be necessary to continue the search. In any case, the search should stop before reaching the *NO* region, for the following reason: positive examples do not need to be repaired; any hypothesis in the *NO* region would cover none of them. Then, top-down learning is bound to produce hypotheses $\mathcal{C}_T$ in the *YES* or in the mushy region.

The above remarks explain why the phase transition constitutes an attractor for top-down learning. Analogous reasons actually hold for any learner without background knowledge.

5.2 Correct identification of the target concept

The second question regards the correct identification of the target concept $\mathcal{C}$. Let us first consider the case where $\mathcal{C}$ belongs to the mushy region. Two cases have been observed and reported in Table 1: either $\mathcal{C}$ involves few literals ($m \leq 6$) and is correctly identified, or the learner retains a number of conjunctive clauses $\mathcal{C}_T$, each covering few positive training examples and performing poorly on the test set.

The reasons why a top-down learner should fail to identify a *long* target concept are illustrated on an example. Let $\mathcal{C}$ be given as:

$$\alpha_0(x_1,x_2) \wedge \alpha_1(x_2,x_3) \wedge \alpha_2(x_2,x_3) \wedge \alpha_3(x_3,x_4) \wedge \\ \alpha_4(x_1,x_4) \wedge \alpha_5(x_1,x_4) \wedge \alpha_6(x_3,x_4) \wedge \alpha_7(x_3,x_4)$$

The corresponding specialization tree, as visited by FOIL, is given in Figure 3. Note that the first literal $\mathcal{C}_1$ can only be selected at random; the information gain cannot provide any useful indication, since all examples have the same number of literals built on every predicate[4]. However, this does not penalize the search here since all predicates are relevant by construction.

Figure 3 develops all possibilities, depending on the first literal chosen. Conditioned by this choice, say $\alpha_0(x_1,x_2)$, literals are sorted on the basis of their information gain. The trouble is that, using this criterion, the choice of the first literal, say $\alpha_6(x_3,x_2)$, *is wrong*; $\mathcal{C}_2$ is not a generalization of $\mathcal{C}$, and the search can thus only wander — or backtrack. In other words, the information gain misleads the search here; even worse, it misleads the search for 7 out of 8 branches. We thus "repare" the choice by forcing the selection of the best correct literal, and proceed again. All oblique arrows in Figure 3 correspond to steps where the information gain misleads the search, and where we thus "repare" the candidate hypothesis $\mathcal{C}_t$, in order to stay in a correct graph. The information gain criterion appears unreliable, until later in the search; in fact, assuming that the first four literals are correct, the information gain successfully leads to the target concept. All in all, top-down learning is bound to fail here, as any path includes several wrong choices in the early steps. The amount of backtrack needed to find any right path is thus enormous.

The above finding can be explained as the information gain relies on the number of models of a candidate hypothesis. But any hypothesis in the *YES* region admits many models in any random example. The number of models associated to any literal is thus hardly meaningful, except when the current hypothesis is close to the target concept. This occurs either when the target concept is short, or when the current hypothesis has been repaired (see the last steps in Figure 3). The variance in the number of models further blinds the selection of literals. Complementary experiments show that the variance reaches its maximum as hypotheses reach the phase transition.

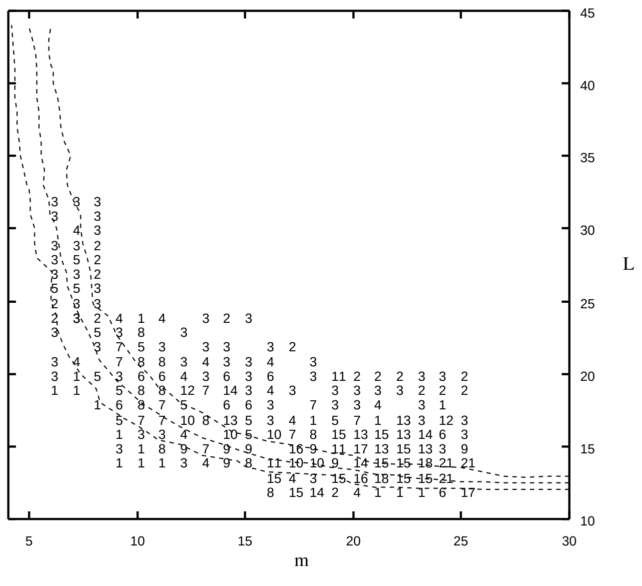


Figure 4. Minimum size of $\mathcal{C}_t$ before the information gain becomes reliable.

[4]In a real-world application, the first literal is selected on the basis of pure attribute-value-like information: the information gain only depends on the number of occurrences of a predicate symbol in positive/negative examples.

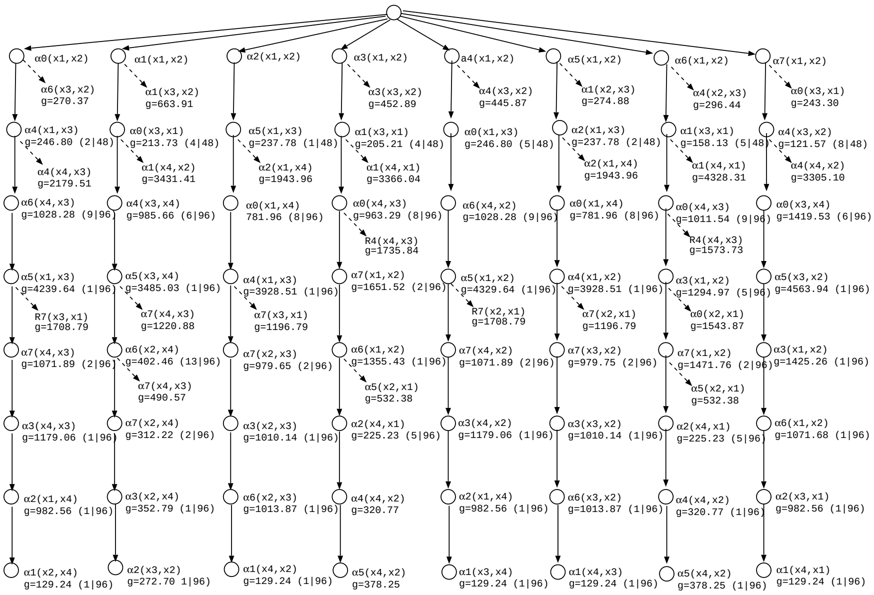


Figure 3. Visiting a specialization tree, when the information gain misleads the search (↘)

The same investigation as in Figure 3 was done for many problems lying within or close to the Phase Transition. Figure 4 reports, at coordinates (m, L), the minimal level t_m of the specialization tree where the information gain becomes reliable. Figure 4 could be thus interpreted as a *reliability map* of the information gain.

Note that, for most problems in the mushy region or on the border between the PT and the NO regions, t_m takes high values, denoting a poor ability to find any correct path; moving farther away from the phase transition, t_c gradually decreases.

5.3 Good approximation of the target concept

According to the above discussion, relational learning is doomed to fail when either the size m of the target concept and/or the number L of constants in the application domain, are high. Still, when both m and L are high (upper right region in Figure 2(a)), FOIL succeedes, and finds highly accurate hypotheses.

This can be explained as follows. Let us assume that the target concept $\mathcal{C}$ belongs to the NO region.

We show that any generalization ξ of the target concept $\mathcal{C}$ will almost surely correctly classify any training or test examples, provided that ξ belongs to the NO region: negative examples are randomly constructed; hence, any hypothesis in the NO region will be correct; in particular, ξ is correct. On the other hand, any example covered by $\mathcal{C}$ is also covered by ξ; this implies that ξ covers all positive examples. Finally, any generalization of $\mathcal{C}$ that belongs to the NO region is complete and (almost surely) correct.

It follows that, if the learning search happens to examine a generalization ξ of $\mathcal{C}$ which is close to the NO region, ξ will be considered an optimal hypothesis, which will stop the search. The success of relational learning, with respect to predictive accuracy, thus depends on the probability of finding a generalization ξ of $\mathcal{C}$ on the edge of the phase transition.

Let m denote the number of literals of $\mathcal{C}$; the number $H(v, m)$ of generalizations of $\mathcal{C}$, with v literals, has been analytically computed, and reaches its maximum at $v = \frac{m}{2}$. This might explain why relational learning succeeds to find accurate approximations of $\mathcal{C}$ when the size m of $\mathcal{C}$ is greater than twice the critical value of m (i.e., the value m_{cr} such that (m_{cr}, L) falls on the crossover curve). As the phase transition contains an exponential number of generalizations of $\mathcal{C}$, there is a reasonable chance to find one; finding one ensures a perfect predictive accuracy on the training and test sets[5].

[5] Another possible interpretation for the fact that it might be easier to approximate larger concepts than shorter ones, is the following. As m increases, more and more modifications are needed to turn a random example into a positive one. This is done by forcing a model of $\mathcal{C}$ in any positive example. The distribution of the positive examples thus becomes increasingly different from the uniform one. However, it is unclear how FOIL directly exploits this information.

6. Conclusion

According to the current trend in Combinatorial Search the actual behavior of an algorithm on real hard-on-average problems, i.e. in the phase transition, conveys more information than its mere worst-case complexity (Hogg et al., 1996).

Following these lines the present paper reports on a systematic experiment, confronting three up-to-date FOL learners to a broad range of artificial learning problems, lying within and outside the phase transition. Despite the simplifications done (uniform distribution of the examples, conjunctive target concept), our experiment might shed some light on the actual limitations of these learners.

The first empirical lesson is that the learning search most often ends up exploring the phase transition; in retrospect, this is hardly surprising, as only concepts/hypotheses in the phase transition can separate the examples (section 2). Incidentally, this implies that FOL learning fully faces the complexity of the covering test, which might raise some doubts as to the scalability issue. This result is supported by the systematic experiments reported here and also by complementary experiments on real-world applications (Giordana & Saitta, 2000).

Second, the widely used information gain criterion appears to consistently mislead the top-down search of a long target concept. The signal-to-noise ratio appears to be quite low in the early learning steps, as any short hypothesis admits a huge number of models in any example. Third, a large "blind spot" appears in the learning landscape; for learning problems (concept/examples) in this region, all three learners consistently fail to provide anything better than random guessing.

Hopefully, these findings might help to reconsider the key issues of FOL learning. Indeed, new learning biases seem to be required to construct scalable FOL learners, and discover more complex concepts than those considered so far (?).
Further research is required to determine a new top-down search criterion, as our experiments suggest that the number of models might be a very noisy indicator. Alternatively, it might be worth reconsidering bottom-up search strategies.
But the primary issue, in our opinion, is to re-design the search space; typically, if all relevant hypotheses lie in the phase transition, then first question is how to explore this particular region.

References

Anglano, C., Giordana, A., Lobello, G., & Saitta, L. (1998). An experimental evaluation of coevolutive concept learning. In *Proceedings of the 15th International Conference on Machine Learning*, pp. 19–23 Madison, WI.

Botta, M., & Giordana, A. (1993). SMART+: A multi-strategy learning tool. In *IJCAI-93, Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, pp. 937–943 Chambéry, France.

Botta, M., Giordana, A., & Saitta, L. (1999). Relational learning: Hard problems and phase transitions. In *Proceedings of th 16th International Joint Conference on Artificial Intelligence*, pp. 1198–1203 Stockholm, Sweden.

De Raedt, L. (1997). Logical setting for concept-learning. *Artificial Intelligence*, *95*, 187–202.

Dietterich, T. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*.

Giordana, A., & Saitta, L. (2000). Phase transitions in relational learning. *Machine Learning*, *x*, in press.

Hogg, T., Huberman, B., & Williams, C. (Eds.). (1996). *Artificial Intelligence: Special Issue on Frontiers in Problem Solving: Phase Transitions and Complexity*, Vol. 81(1-2). Elsevier.

King, R., Srinivasan, A., & Stenberg, M. (1995). Relating chemical activity to structure: an examination of ILP successes. *New Generation Computing*, *13*.

Muggleton, S. (1995). Inverse entailment and PROGOL.. *New Gen. Comput.*, *13*, 245–286.

Muggleton, S., & De Raedt, L. (1994). Inductive logic programming: Theory and methods. *Journal of Logic Programming*, *19*, 629–679.

Quinlan, R. (1990). Learning logical definitions from relations. *Machine Learning*, *5*, 239–266.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, *14*, 465–471.

Sebag, M., & Rouveirol, C. (1997). Tractable induction and classification in first order logic via Stochastic Matching. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence*, pp. 888–893 Nagoya, Japan.