Coping with exceptions in multiclass ILP problems using possibilistic logic

Mathieu Serrurier
UPS, IRIT, 118 route de Narbonne,
31062 Toulouse, France
serrurie@irit.fr

Henri Prade
UPS, IRIT, 118 route de Narbonne,
31062 Toulouse, France
prade@irit.fr

Abstract

The handling of exceptions in multiclass problems is a tricky issue in inductive logic programming (ILP). In this paper we propose a new formalization of the ILP problem which accounts for default reasoning, and is encoded with first-order possibilistic logic. We show that this formalization allows us to handle rules with exceptions, and to prevent an example to be classified in more than one class. The possibilistic logic view of ILP problem, can be easily handled at the algorithmic level as an optimization problem.

1 Introduction

The handling of exceptions is a serious bottleneck in ILP due to the formalization in first-order logic. Indeed, when a rule has some exceptions, i.e. some examples are misclassified by it, there is no way to compensate these exceptions by means of another rule. So, a hypothesis accumulates all the exceptions of the rules that appear in it. Moreover, when dealing with more than two classes, a rule with some exceptions may prevent another one to perform the right classification. A proper handling of exceptions can be made by adding to the standard ILP setting a logical constraint that expresses that an example can be only classified into one class. Thus, having some exceptions may lead to inconsistency. In contrast with first-order logic, default reasoning encoded in possibilistic logic is well-suited for dealing with inconsistency. In this paper, we extend the ILP settings in a possibilistic way in order to handle inconsistency due to exceptions.

2 Preliminaries

Stated in the general context of first-order logic, the task of ILP is to find a non-trivial set of formulas H such that $B \cup H \models E$ given a background theory B and a set of examples E of the form $C(x, y)$ where x denotes the identification key of an example and y a class. E, B and H here denote sets of Horn clauses. In order to treat multiple classification of an example as inconsistency, we reformulate the ILP problem, by adding the following classification constraint $D \equiv \forall X, Y, Z\ C(X, Y) \cup C(X, Z) \rightarrow Y = Z$, as follows : given B, D and E, the goal is to find H such as $B \cup D \cup H \models E$.

We extend propositional possibilistic logic [Dubois *et al.*, 1994] to the first-order case. Possibilistic logic is sound and complete for refutation, using an extended resolution rule, with respect to a semantics in terms of a complete plausibility preorder on the interpretations (encoded by a possibility distribution) [Dubois *et al.*, 1994]. S denotes a set of Herbrand interpretations. A possibility distribution can be defined on Herbrand interpretations as well. The possibilistic degree of a formula ϕ is the maximum possibility level of its models and is denoted $\Pi(\phi)$. Necessity is the dual notion of possibility. It refers to the possibility degrees of the counter-models of a formula :$N(\phi) = 1 - \Pi(\neg\phi)$. A possibilistic first-order formula is a pair (ϕ, α) where ϕ is a first-order formula and $\alpha \in]0, 1]$. It is understood as a constraint on an unknown necessity measure of the form $N(\phi) \geq \alpha$. Given a set K of possibilistic formulas, the α-cut of K is $K_\alpha = \{\phi | (\phi, \beta) \in K, \beta \geq \alpha\}$. Given T a set of classical first-order formulas, T is minimal w. r. t. a formula ϕ iff $T \models \phi$ and $\forall \psi \in T, T - \psi \not\models \phi$. The following definition avoids the drowning problem (i.e., consequences are lost when their implicants are taken in inconsistent sets of formulas, although they don't contribute to inconsistency) by checking if it exists a proof of ϕ in K_α, which is free from inconsistencies at level α.

Definition 1 *Given K a set of possibilistic first-order formulas, $K \models_\pi (\phi, \alpha)$ iff $\exists K' \subset K_\alpha, K' \not\models \bot$ such as K' minimal w. r. t. ϕ and $\nexists K''$ minimal w. r. t. $\bot$ such that $K' \subset K'' \subseteq K_\alpha$.*

In practice, this definition makes sure that, if two logical consequences are inconsistent, the one which has the highest necessity degree is preferred. As already advocated in the introduction, we propose to use possibilistic logic in ILP for a better handling of exceptions.

3 Possibilistic ILP

Given H a standard ILP hypothesis, let N_H denote a function (called priority function) which at each rule in H associates a priority level, to be understood as a necessity degree. This gives birth to a possibilistic hypothesis $H_p = \{(h, N_H(h)); h \in H\}$. Let B_p and D_p be composed by all formulas that appear respectively in B and D, with 1 as necessity level. The priority function N_E over the examples is deduced from H_p as follows :

$$e \in E, N_E(e) = \begin{cases} 1, \text{ by convention, if } \nexists \alpha > 0 \\ \text{ such that } D_p \cup B_p \cup H_p \models_\pi (e, \alpha) \\ \\ max\{\alpha > 0, D_p \cup B_p \cup H_p \models_\pi (e, \alpha)\}, \\ \text{ otherwise.} \end{cases} \quad (1)$$

Then, given D, B and E, the goal of possibilistic ILP is to find H_p, composed by a classical hypothesis H and an associated priority function N_H such that $B_p \cup D_p \cup H_p \models_\pi E_p$ with $E_p = \{(e, N_E(e)); e \in E\}$. Note that, if it exists such H_p, this hypothesis will be correct and complete. This enlarges the scope of classical ILP by learning sets of default rules in a framework that handles exceptions, the one of possibilistic logic.

Any possibilistic hypothesis, even it contains some rules with exceptions, can be completed in order to be correct and complete by adding the misclassified examples with 1 as necessity level. Then, the possibilistic ILP problem can be reformulated as an optimization problem :

Given D, B and E, the goal of possibilistic ILP is to find H_p that maximizes the accuracy (i.e. the proportion of well-classified examples). This definition of possibilistic ILP problem is fully in agreement with the paradigm of the minimization of the empirical risk. Since here, necessity levels are only used for obtaining an ordering of the formulas in H, this induces equivalence classes of priority functions N_H. Two priority functions N_H and N'_H, belong to the same equivalence class, i.e. $N_H \equiv N'_H$ if and only if $\forall h1, h2 \in H$ if $N_H(h1) > N_H(h2)$ then $N'_H(h1) > N'_H(h2)$.

Proposition 1 *Given H, finding the class of equivalence of priority functions such the accuracy is maximal is NP-complete with respect to the number of formulas in H.*

It shows that, although using a possibilistic rather than the classical setting is always more effective, finding the best priority function over formulas in H may be computationally very costly. Note that choosing a particular ordering based on the confidence or the support degrees of the rules is not optimal in general. It suggests to use heuristics for inducing hypotheses together with their priority function.

4 Experimentations

	A	B	C	D	E	$\sum$
Maximum	52	38	28	57	89	277
Foil	17	5	7	9	5	43
Indigo	21	14	9	18	33	95
Cilgg	26	12	10	18	37	103
Pilp avg	23 [0]	11.6 [1.3]	9.5 [0.5]	18.3 [3.6]	55.3 [1.5]	117.8 [5.03]
Pilp max	23	13	10	23	57	125

In order to learn possibilistic logic rules, we learn a hypothesis directly together with its priority function. The hypotheses are learnt by using a stochastic exploration of the hypothesis space as explained in [Serrurier *et al.*, 2004]. The priority is found by using a greedy algorithm based on the 2-opt method (switching the order of rules two by two while accuracy increases). Our algorithm is denoted as Pilp. Since Pilp is a non deterministic algorithm, the results that are presented are average results on 50 running steps with the same settings. The value in the brackets are standard derivations for the 50 results. For each experiment, two results are shown : Pilp avg is the average result for the 50 tests, and Pilp max is the best result found in the 50 tests. The experimentation is made with the finite element MESH Design dataset because it represents a typical hard ILP multiple class problem. It describes the structure of a mesh with unary and binary predicates. The dataset contains 277 examples that describe 13 classes. The dataset is split in 5 sub datasets denoted by A, B, C, D and E. The test of the accuracy of the algorithm is made by testing on one subset a hypothesis induced from the other sub datasets. Results are shown in the previous table. The results for the algorithms other than Pilp can be found in [Kietz, 2002]. The results are clearly in the favor of Pilp algorithm which increases the number of examples covered by the most effective algorithm up to 10% in average and 20% at maximum.

5 Conclusion

In this paper, we have proposed a new ILP formalization for dealing with exceptions in a multiple class problem. In order to do that, we have extended possibilistic propositional logic to a first-order setting. Then, by treating exceptions as inconsistency, we have reformulate the ILP problem in first-order possibilistic logic. In this reformulation, the ILP problem is turned in an optimization problem. This formalization allows our algorithm to learn sets of default rules. In this context, it may exist a correct and complete hypothesis which contains rules that have some exceptions. Possibilistic ILP is more flexible and more general than first-order decision lists [Mooney and Califf, 1995] and allows us to correctly cope with recursive hypotheses. Experiments have proved that an implementation of possibilistic ILP may be very effective for propositional or for relational learning and can compete with the best machine learning algorithms.

References

[Dubois *et al.*, 1994] D. Dubois, J. Lang, and H. Prade. Possibilistic logic. In D. Gabbay et al., editor, *Handbook of Logic in Artificial Intelligence and Logic Programming, vol. 3*, pages 439–513. Oxford University Press, 1994.

[Kietz, 2002] J. U. Kietz. Learnability of description logic programs. In S. Matwin and C. Sammut, editors, *Proceedings of the 12th Inter. Conference of Inductive Logic Programming, ILP 2002*, pages 117–132. Springer, 2002.

[Mooney and Califf, 1995] R. Mooney and M. E. Califf. Induction of first-order decision lists: results on learning the past tense of english verbs. *Journal of Artificial Intelligence Research*, 3:1–24, 1995.

[Serrurier *et al.*, 2004] M. Serrurier, H. Prade, and G. Richard. A simulated annealing framework for ilp. In R. Camacho, R. King, and A. Srinivasan, editors, *Proceedings of ILP 2004*, pages 288–304. LNAI 3194, Springer, 2004.