Proc. FGML-95, Annual Workshop of the GI Special Interest Group Machine Learning (GI FG 1.1.3), ed. K. Morik and J. Herrmann, Research Report 580, Univ.Dortmund, 1995.

The ILP description learning problem: Towards a general model-level definition of data mining in ILP

Stefan Wrobel Saso Džeroski*
GMD, FIT.KI
Schloß Birlinghoven
53754 Sankt Augustin
stefan.wrobel@gmd.de, saso.dzeroski@gmd.de

Abstract

The task of discovering interesting regularities in (large) sets of data (data mining, knowledge discovery) has recently met with increased interest in Machine Learning in general and in Inductive Logic Programming (ILP) in particular. However, while there is a widely accepted definition for the task of concept learning from examples in ILP, definitions for the data mining task have been proposed only recently. In this paper, we examine these so-called "non-monotonic semantics" definitions and show that non-monotonicity is only an incidental property of the data mining learning task, and that this task makes perfect sense without such an assumption. We therefore introduce and define a generalized definition of the data mining task called the ILP description learning problem and discuss its properties and relation to the traditional concept learning (prediction) learning problem. Since our characterization is entirely on the level of models, the definition applies independently of the chosen hypothesis language.

1 Introduction

The task of discovering interesting regularities in (large) sets of data (data mining, knowledge discovery) has recently met with increased interest in Machine Learning in general and in Inductive Logic Programming (ILP) in particular. In data mining, one is typically given one or more data sets and is interested in the properties that *describe* these data, i.e., that are true of the data. In this learning problem, which

*ERCIM-fellow at GMD, on leave from Jozef Stefan Institute, Artificial Intelligence Laboratory, Ljubljana, Slovenia.

we will refer to as the *description learning problem*, the goal thus is to find out more about the data, not to replace them. In contrast, in the classical "concept learning from examples" task, one is given background knowledge and sets of positive and negative examples and is interested in finding a hypothesis that allows to *predict* the examples based on the background knowledge. In this learning problem, which we will refer to as the *prediction learning problem*, the goal is to find a generalized hypothesis which can replace the examples.

For the prediction learning task, a widely accepted general definition has emerged that can be found almost identically in most ILP papers addressing this task [Mug91, MDR94, LD94]. On the other hand, existing work on defining the description learning task has concentrated exclusively on the so-called "non-monotonic semantics" of ILP [RB93] [DRD94, Dž95], strongly associating the task of finding descriptive properties of data with a closed-world interpretation of the data in the spirit of Helft's formalization [Hel89]. In this paper, we examine whether these restrictions can be relaxed and propose a general definition of the description learning task which admits both an open world and a closed world instantiation, showing that the non-monotonic interpretation of examples is not a central characteristic of the description problem. We discuss the properties of this generalized definition and describe its relation to the prediction learning problem. Since our characterization is entirely on the level of models, our definition applies independently of the chosen hypothesis language.

The paper is organized as follows. In section 2, we first illustrate the description learning problem with an example that will be used throughout the paper. In section 3, we then briefly recapitulate the existing non-monotonic ILP semantics definitions of De Raedt and Bruynooghe [RB93] and of Dzeroski [Dž95]. In section 4, we generalize these existing definitions by introducing a general "description" relationship between a hypothesis and a set of data, encompassing both a closed-world and an open world interpretation of the data. In section 5, we then use examples to discuss the requirements that we think the description relationship must meet, and present and discuss different instantiations of this relationship. Based on these instantiations of the description relationship, we can then formally capture the relationship between the description problem and the prediction problem (section 6. We then discuss the problem of assigning a degree of confirmation to hypotheses, looking at a number of implemented systems that address variants of the description learning task (section 7), and then attempting a general model-level definition of degree of confirmation (section 8). The final section contains our conclusions (section 9).

2 An example of the description problem

In the simple case of the description problem, we might have only a single body of data to describe [RB93], i.e., we are interested in all properties that are true of this body of data. In a more general case, it can be advantageous to think of

the given data set or theory as describing a number of different objects, situations, or cases in only some of which we are interested [Dž95]. We would then like the learning hypothesis to be such that it describes the interesting cases, but none of the uninteresting cases, i.e., we want general properties that are true of simply all cases to be filtered out.

As an illustrative example, assume we are given a database of small families and want to learn about their properties. The database is expressed with the predicates

father(<child >,<father >).
mother(<child >,<mother >).
married(<husband >,<wife >).

and consists of two groups of families. We are interested in the first group, consisting of the "j"- and "t"-families described in data sets d_1^+ and d_2^+, and not interested in the second, consisting of the "m"-family described in data set d^-:

d_1^+ :	d_2^+ :	d^- :
father(jim, john).	father(tim, tom).	father(mike, marty).
father(jane, john).	father(tanja, tom).	father(marcia, marty).
mother(jim, jill).	mother(tim, tammy).	mother(mike, mary).
mother(jane, jill).	mother(tanja, tammy).	mother(marcia, mary).
married(john, jill).	married(tom, tammy).	not(married(marty, mary)).

Here, one possible descriptive hypothesis that we might be interested in is

h_1: father(X,Y) & mother(X,Z) $\rightarrow$ married(Y,Z).

telling us that in our first group of families, people who have children together are married. In contrast, we would not be interested in the property

father(X,Y) & mother(Z,Y) $\rightarrow$ fail.

since it true of all families (noone can be a father and a mother at the same time).

3 The non-monotonic semantics definitions

This section reviews several definitions of the so-called nonmonotonic semantics of ILP, which are closely related to the description learning problem. The definitions are based on model theory notions from standard first-order logic. We therefore briefly introduce the key notions of interpretations and models. We assume familiarity with the syntax of first-order logic [Llo87].

An interpretation comprises a mapping from terms to domain objects (called a pre-interpretation) and a mapping from predicates to domain relations, for a particular (real-life) domain under study. Taken together, these two mappings assign a truth value to each statement in the particular first-order language. As the truth values of more complex statements (involving logical operations and quantifiers)

can be determined from the truth values of ground atomic statements by following a fixed set of rules, we can identify an interpretation by the set of ground atomic statements that it makes true.

An interpretation is a model for a particular statement if it makes the statement true, i.e., if the statement is true in the interpretation. A theory is a set (conjunction) of first-order statements. An intepretation that makes true each statement of a theory is called a model of the theory.

A particular kind of models are Herbrand models, which are Herbrand interpretations. Herbrand interpretations have a trivial pre-interpretation that maps each term to itself. Of special interest are minimal Herbrand models: a minimal Herbrand model of a first-order theory contains only the ground facts that are a logical consequence of (follow from) the theory. The minimal Herbrand model of theory T will be denoted by $\mathcal{M}^+(T)$. It is uniquely determined if T consists of definite clauses, but need not be so otherwise.

Definition 1 (Global nonmonotonic ILP) *For a hypothesis language L_H, the following problem is called the* global nonmonotonic ILP problem.
Given:

- *a dataset D in the form of a definite clausal theory (a set of definite clauses)*

Find:

- *a hypothesis H expressed in the hypothesis language L_H*
- *such that H is true in $\mathcal{M}^+(D)$,*
- *H is complete, i.e., for all $H' \in LH$, if H' is true in $\mathcal{M}^+(D)$ then $H \models H'$, and*
- *H is minimal (non-redundant), i.e., for all $H' \subset H$, $H' \not\models H$.*

The above definition was introduced by De Raedt and Bruynooghe [DRB93], In this definition, a dataset D is given, and the task is to find a hypothesis H that is true in the minimal Herbrand model of D. Using $\mathcal{M}^+(D)$ corresponds to making a closed-world assumption, which is a form of nonmonotonic reasoning, hence the name nonmonotonic semantics of ILP. Related nonmonotonic ILP semantics have been proposed by Helft [Hel89] and Flach [Fla92].

The above definition assumes a single set of data D. The hypothesis H should capture the properties of D as a whole. A typical dataset would comprise facts on all families in a given region. Background knowledge can also be provided as a part of the dataset in the form of a set of definite clauses.

Studying the computational complexity of learning within the above setting, Džeroski [Dž95] proposes a generalized definition of nonmonotonic ILP. In this definition, one is interested in the properties common to several datasets of interest, possibly properties that distinguish between the datasets of interest and other given

datasets considered not to be interesting. The validity of properties (statements) is tested locally, i.e., on each dataset separately. In the family case, a single dataset comprises the facts about one family. The use of background knowledge that is common across all datasets is also allowed. Special cases of this setting are considered by De Raedt and Džeroski [DRD94] and De Raedt and Van Laer [DRVL95].

Definition 2 (Local nonmonotonic ILP) *For a hypothesis language L_H, the following problem is called the* local nonmonotonic ILP problem.
Given:

- *a set of positive examples E^+ (datasets of interest),*
- *a set of negative examples E^- (uninteresting datasets),*
- *and background knowledge B, all in the form of definite clausal theories,*

Find:

- *a hypothesis H expressed in the hypothesis language L_H*
- *such that for each positive example $D^+ \in E^+$, H is true in $\mathcal{M}^+(D \wedge B)$,*
- *and for each negative example $D^- \in E^-$, H is false in $\mathcal{M}^+(D \wedge B)$.*

4 A general model-level definition

A key element of the above non-monotonic semantics definitions of the ILP description learning task is that the datasets (examples) are interpreted as models, i.e., as the set of statements from their Herbrand bases that they imply. In the case when datasets comprise ground facts only, this means that all facts that are not explicitly stated are assumed to be false. An alternative way of looking at this is that the datasets that are given are first completed using a closed-world assumption, which is a form of non-monotonic reasoning) and then used for checking hypotheses on them (hence the name non-monotonic semantics).

However, a non-monotonic interpretation of the given data seems appropriate only in circumstances where we can be sure that our datasets are complete or at least completely describe the objects that are mentioned. In many situations where we want to describe a given data set, this data set can be incomplete — information about entities in the dataset might be missing (e.g. because it has not been collected yet) or certain entities might be missing entirely (e.g., future events or parts of the domain that have not been input yet). Since learning is often used during knowledge base construction, this situation can occur quite frequently.

If we want to drop the non-monotonic interpretation of the given data set, this means that unstated statements might actually be true or false. When checking whether a hypothesis describes such a data set, the question arises how to treat such unknown facts. Clearly, we can take an optimistic view and assume truth

values for them just in the way we need to make the hypothesis true, or we can be pessimistic about them and assume their truth values are such that the hypothesis is false.

As an example, assume that in our family problem introduced in section 2 we were given another family (the l-family), but we do not know (yet) whether the mother and the father are married:

d_3^+ :
father(luke, larry).
father(louise, larry).
mother(luke, linda).
mother(louise, linda).

Now if we consider the hypothesis

h_1: father(X,Y) & mother(X,Z) $\rightarrow$ married(Y,Z).

again, if we optimistically assume that

married(larry,linda)

is true, we can accept h_1. However, if we are pessimistic and assume that this fact is false, we must reject h_1.

So clearly, if we drop the closed-world assumption, whether we can accept a certain hypothesis as correctly describing our data depends on the assumptions we make about unknown statements. On the level of models, this can be seen as using a given data set as a partial model, and extending it with additional assumptions into a larger model which is then used to verify the hypothesis. This also clearly shows the relationship to the nonmonotonic setting, which is a special case: there, the partial model given by the data set (or its minimal model) must not be extended at all, so the nonmonotonic semantics forces us to take the most conservative view.

In the general case, whether a hypothesis describes a set of data thus depends on how we extend the data (its model) by additional assumptions. Depending on which kinds of extensions we allow, we get different solutions to the ILP description task. In the following general definition, we have therefore hidden this process in a generic *description relationship* $\models_D$ between different data sets and a hypothesis. In section 5, we will then discuss different instantiations of $\models_D$ and their properties.

Definition 3 (ILP description problem) *For a background knowledge language L_B, an example language L_E, a hypothesis language L_H, an entailment relationship $\models$, and a description relationship $\models_D$, we call the following problem the* ILP description learning problem $ILP_D(L_B, L_D, L_H, \models, \models_D)$*:*
Given:

- *background knowledge B expressed in the background knowledge language L_B*
- *interesting datasets D^+ expressed in the dataset language L_D*

- *uninteresting datasets* D^- *expressed in the dataset language* L_D
- *such that* B *is consistent with* D^+ *and* D^- $(B \cup D^+ \cup D^- \not\models \Box)$[1]

Find:

- *a learning hypothesis* H *expressed in the hypothesis language* L_H
- *such that* H *describes* B *and* D^+*, but not* D^- $((B, D^+) \models_D H, (B, D^-) \not\models_D H)$.

Often, we also require that

- H *is complete, i.e., for any other descriptive hypothesis* H' *in* L_H*,* $H \models H'$*, and*
- H *is non-redundant, i.e., for any proper subset* $H' \subset H$*,* H' *is not complete.*

Note that if D^- is not considered (formally, $D^- = \{\Box\}$, i.e., a single inconsistent set without models), and D^+ consists of a single set only, the above task turns into the simple case of the description problem, i.e., simply finding all properties true of a set of data.

5 The description relationship $\models_D$

As stated above, the basic intuition behind the description relationship on an individual dataset/theory Γ is that by making assumptions about unknown statements, a model of Γ can be found in which a hypothesis is true in a standard logical sense.

Definition 4 (Description relationship on individual datasets)
A hypothesis H *is satisfiable in a dataset/theory* Γ $(\Gamma \models_D H)$ *if and only if* Γ *has a model in which* H *is true, i.e.,* Γ *and* H *are consistent:*

$$\Gamma \models_D H \text{ iff } \Gamma \cup H \not\models \Box$$

This is the most general definition, as it allows any model of a given dataset to be used, i.e., it does not limit the assumptions one can make in any way. We will postpone the discussion about how such assumptions should be limited to a later section (section 7), but it should be clear that without limits uninteresting hypotheses will be admitted.

Instead, let us first discuss how the description relationship should be extended from single datasets to sets of datasets as provided for in our general definition. Should we require that a hypothesis describe each individual data set, or all of them together? Let us consider some example hypotheses in our family problem to answer this question.

[1] Alternatively, we could simply require that $B \cup D^+ \not\models \Box$, i.e., allowing arbitrary hypothetical cases as uninteresting datasets that need not be consistent.

h_1: father(X,Y) & mother(X,Z) → married(Y,Z).
h_2: father(X,Y) & mother(U,Z) → married(Y,Z).

Here, for all of d_1^+, d_2^+ and d_3^+, there are models which make h_1 true, and there is no model of d^- which makes h_1 true. In fact, there is also a model of the union of d_1^+, d_2^+ and d_3^+ in which h_1 is true, and this model can simply be the union of the models chosen individually in each d_i^+ to make h true.

In constrast, while the same individual models can be used to make h_2 true on each of the d_i^+, h_2 is not true in the union of these models taken as a model of the union of the d_i^+: the minimal model of $d_1^+ \cup d_2^+ \cup d_3^+ \cup h_2$ must also contain

married(john,tammy)
married(john,linda)
married(tom,jill)
married(tom,linda)
married(larry,jill)
married(larry,tammy)

Thus, to make h_2 true "globally", additional assumptions must be made that are not necessary to make h_2 true locally. Even if h_2 might seem uninteresting, there are of course other "global" hypotheses which could be interesting, e.g.

h_3: married(X,Y) & married(X,Z) → Y=Z

(This hypothesis also describes d^-, however, so we would not be interested in it in this example.)

For h_1, h_2 and h_3, we thus need different kinds of model extensions to make them true globally. Of course we can also find hypotheses where this is entirely impossible. Consider

d_4^+:
p(a). q(a).
p(b).
r(c). not(q(c))

and

d_5^+:
p(1). q(1).
r(b).
r(2). not(q(2)).

and an inductive hypothesis consisting of two clauses:

h_4: {p(X) → q(X). r(X) & q(X) → fail}

Both d_4^+ and d_5^+ have models which make h_4 true, but these two models have incompatible truth assignments to q(b) (true in the first and false in the second model).

It is interesting to note that if we limit ourselves to minimal models (non-monotonic setting), the problem with h_4 cannot occur: As long as the individual datasets are consistent, all local models will be consistent since they are all minimal models. Of course, their union need not be a minimal model of the union of the datasets, so it still makes sense to speak of "local" and "global" hypotheses.

Let us define the notions of local and global truth on a set of datasets more precisely.

Definition 5 (Description relationship on sets of theories)
Let $D = \{d_1, \ldots, d_n\}$ be a set of datasets/theories, B background knowledge and H a statement (inductive hypothesis). We define three description relationships as follows:

Locally descriptive $(B, D) \models_D^l H$ *iff for all* $i \in \{1, \ldots, n\} : B \cup d_i \models_D H$.

Globally descriptive $(B, D) \models_D^g H$ *iff* $B \cup \bigcup_{i \in \{1,\ldots,n\}} d_i \models_D H$.

Globally descriptive with local models $(B, D) \models_D^{lg} H$ *iff* $\forall i \in \{1, \ldots, n\}$ *there is a local model* M_i *of* $B \bigcup d_i$ *and* $\bigcup_{i \in \{1,\ldots,n\}} M_i$ *is a model of* $B \cup \bigcup_{i \in \{1,\ldots,n\}} d_i$.

The important point in the global with local models definition is that the models must be based on the individual datasets and then combined together. This means that the set of true facts must be based on the vocabulary of individual dataset, and that all facts that "mix" objects from different datasets are false (since they are not in the Herbrand universe of any dataset, they cannot be true in the model of any dataset, and thus are not included in the union). This is in fact a "partial" minimal model approach that minimizes only the "cross-dataset" part of the model.

Let us now consider the relationships between the different definitions of the description relationship given above. For a single dataset, all three definitions coincide, since the union of all datasets is simply that single dataset. In the general case, the relations are somewhat more complex. Let us first look at the relationship between the third and the first two definitions.

Proposition 1 *Let D be a set of datasets/theories, B background knowledge and H a statement (an inductive hypothesis).*

- *If* $(B, D) \models_D^{lg} H$ *then* $(B, D) \models_D^g H$.
- *If* $(B, D) \models_D^{lg} H$ *then* $(B, D) \models_D^l H$.

Proof: Whenever $(B, D) \models_D^{lg} H$, the union of local models is a model of the union of the d_i, so $(B, D) \models_D^g H$. The local models ensure that $(B, D) \models_D^l H$. □

Note that in the global with local models definition it would not be sufficient to require that the hypothesis simply be true in the union of local models (without

requiring it to be a model of the union). If this condition were dropped, definitions 2 and 3 would be incomparable, since even if a statement is true in the union of individual models, it might no longer be true if these models are extended into a model of the union of the statements. Consider the two datasets {p(a)} and {p(X) → q(X)}. Choose {p(a)} as a model of the first and ∅ as a model of the second. not(q(a)) is true in both of these models and in their union, but is not true in any model of the union of the two datasets.

The converse of this proposition is not true, as evidenced by our examples above. According to the local definition, the hypotheses h_1 to h_4 are considered descriptive, according to the global definition, h_4 is not admitted any more, and according to the global with local models definition, h_2 is not admitted any more. Thus, the third definition is stronger than both the local and the global definition.

The local and the global definition in turn, however, are incomparable in the general case. Consider the following datasets.

$d_6^+ = \{\mathsf{p(a).p(X)} \rightarrow \mathsf{q(X,Y).not(q(a,a)).}\}$
$d_7^+ = \{\mathsf{q(a,b).}\}$

and the hypothesis

$h_5 := \mathsf{p(X)} \rightarrow \mathsf{q(X,Y)}.$

(which actually is part of d_6^+). For $d_6^+ \cup d_7^+$, we can find a model that makes h_5 true, for example the model {p(a).q(a,b).}. However, it is impossible to find a model of d_6^+ due to its limited alphabet. The only q-fact in its Herbrand universe is q(a,a) which is required to be false, so we cannot make h_5 true. Thus, even though there is a model of the union of all d_i^+, when we restrict this model to the local alphabets it is no longer a model.

Fortunately, if we restrict ourselves to the language class of range-restricted clauses, the above cannot happen any more.

Proposition 2 *Let D be a set of datasets/theories, B background knowledge and H a statement (an inductive hypothesis), all in the language of range-restricted clauses.*

- *If $(B,D) \models_D^g H$ then $(B,D) \models_D^l H$.*

Proof: Assume we have a model of the union of all the d_i's. When restricting it to a single d_i, all statements not in the Herbrand universe of d_i are not in the domain/range of the model any more, so if one of them is needed to make a statement in d_i true, there would be a problem. However, for atomic statements in d_i, this cannot happen since clearly they remain in the Herbrand universe of d_i. For clauses, if one of the premise statements is dropped by the restriction to a local model, the clause will still be true. Since in the restricted model, all premise instantiations must be part of the local model, and the clause is range restricted, all conclusion instances must be part of the local model also, so they also will not be dropped by the restriction to a local model. □

The three definitions all seem appropriate under certain circumstances. The local definition seems appropriate if the datasets are interpreted as describing different worlds, not different parts of the same world. In such a scenario, we would not be interested in joining the datasets together (in fact, they need not even be consistent), so we would not worry about choosing consistent models. In most situations, however, we would think of our datasets as "cases" that belong to one global world, so the global definition would be appropriate. Finally, the global with local models definition is useful if we want to prevent the learning system from connecting different cases in our knowledge base, since it admits only those hypotheses which can be made true by making assumptions about individual cases.

Finally, for the uninteresting data sets D^-, the choice of $\models_D$ has the opposite effects: if a hypothesis does not describe a dataset under the local definition of $\models_D$, it does not describe the dataset under the other two definitions of $\models_D$ either. Thus, it could be quite sensible to require that a hypothesis does not describe the datasets from D^- under the local definition of $\models_D$, while requiring it to describe the datasets from D^+ under one of the global definitions of $\models_D$.

6 Relating the description and the prediction problem

Given the above definitions of when a hypothesis describes a set of data, we can now relate the description learning problem to the prediction learning problem. The important point to consider is how to map the sets of positive and negative examples to the components of the description learning problem, so let us recall the exact definition of the prediction problem first [KD94].

Definition 6 (ILP prediction learning problem) *For a background knowledge language L_B, an example language L_E, a hypothesis language L_H, and an entailment relationship $\models$, we call the following problem the* ILP prediction learning problem *$ILP_P(L_B, L_E, L_H, \models)$:*
Given:

- *background knowledge B expressed in the background knowledge language L_B*
- *positive examples E^+ expressed in the example language L_E*
- *negative examples E^- expressed in the example language L_E*
- *such that B is consistent with E^+ and E^- ($B \cup E^+ \cup E^- \not\models \square$),*

Find:

- *a learning hypothesis H expressed in the hypothesis language L_H*
- *such that H together with B entails the positive examples ($H \cup B \models E^+$)*

- *and does not entail any of the negative examples, i.e., for all* $e^- \in E^-$: $H \cup B \not\models e^-$*).*

The important comparison we must make is in the conditions imposed on E^+/E^- and D^+/D^-, respectively. The prediction problem imposes a stronger requirement on H with respect to E^+ than the description problem with respect to D^+, whereas the description problem imposes a stronger requirement on H with respect to D^- than the prediction problem with respect to E^-.

Proposition 3 *Let* $ILP_P(L_B, L_E, L_H, \models)$ *be an ILP prediction learning problem, and* $ILP_D(L_B, L_D, L_H, \models, \models_D)$ *an ILP description learning problem such that* $L_D = L_E$*. Let* P *be an instance of* $ILP_P(L_B, L_E, L_H, \models)$ *with* B*,* E^+ *and* E^-*, and* D *be an instance of* $ILP_D(L_B, L_D, L_H, \models, \models_D)$ *with* B*,* D^+*, and* D^-*.*

1. *Any solution* H *of* P *is (more specific than) a solution to* D *(under the local and global description relationship) if* $D^+ := E^+$ *and* $D^- := \{\{\Box\}\}$*.*

2. *Any solution* H *of* D *(under the local description relationship) is a solution to* P *if* $E^- := D^-$ *and* $E^+ := \emptyset$*.*

Proof: 1. From the definition of ILP_P, we know that $H_P \cup B \models E^+$. This means that every model of $H_P \cup B$ is also a model of E^+. Thus, there is a model of $H_P \cup B \cup E^+ = H_P \cup B \cup D^+$. Thus, H_P is a descriptive hypothesis of $B \cup D^+$, cannot be true of D^-, and thus must be entailed by any solution to the description problem. 2. From the definition of ILP_D, we know that $(B, D^-) \not\models_D H$. In the local definition of $\models_D$, this means that for all d^- in D^- $B \cup H \cup d^- \models \Box$, i.e., there is no model of $B \cup H$ in which any element of $D^- = E^-$ would be true, so in particular no element of E^- is entailed by $B \cup H$. Since E^+ is empty, the prediction conditions holds vacuously. $\Box$

Note that if we had mapped E^- to D^- in (1), the above proposition would not hold: even though we know that $B \cup H \not\models e^-$ for all $e^- \in E^-$, this does not mean that there could not be models of $B \cup H$ in which the negative examples are true, i.e., if we mapped E^- to D^-, we could not be sure that a predictive hypothesis H would not describe D^-. In summmary, the prediction learning problem has a stronger requirement on the hypothesis with respect to the positive examples (entailment instead of description) and a weaker requirement on the hypothesis with respect to the negative examples (non-entailment instead of non-description).

7 Degree of confirmation

Given the three possible ways of defining the description relationship on sets of theories, let us return to the question that we had postponed earlier on, namely what restrictions to place on the description relationship with respect to individual datasets. As pointed out above, if we interpret our datasets as partial models that

can be extended by additional assumptions, we certainly do not want to permit arbitrary (and arbitrarily many) assumptions to be made — we could then make any hypothesis descriptive of any dataset that does not directly contradict it. In particular, this would hold true even for empty datasets.

Intuitively, there should be some difference in the degree of descriptiveness or truth between a hypothesis that is true only in a model consisting of "almost only" assumptions and a hypothesis that is true with "almost no" additional assumptions. In our example, we can compare h_1 and h_2 in this respect based on the three families d_1^+, d_2^+ and d_3^+ (sections 2 and 4). The reader will recall that to make h_1 true on all three datasets (families), only one additional assumption was needed, married(larry,linda) (section 4). For h_2, on the other hand, we needed this assumption and six additional "global" assumptions (section 5). So somehow, as h_1 needs fewer assumptions, it describes the data better.

How can we precisely define this notion of better fit to the data? Ideally, we would like some measure which would allow us to compare different hypotheses and e.g. accept only those that are above a user-specified threshold. Here, it could be instructive to look at the solutions that existing algorithms for the description learning task in ILP have chosen with respect to their particular hypothesis language.

7.1 RDT

Learning task. RDT ("Rule Discovery Tool", [KW92]) solves the ILP_D problem in a generalized function-free Horn program representation, i.e., using clauses with a single head, but which can use negated literals in both the head and the body of the clause. Background knowledge can consist of arbitrary sets of statements in this representation, but is transformed into sets of ground literals by depth-limited forward inference (through the inference engine of MOBAL of which RDT is one part). RDT solves the simple case of the description learning problem, i.e., its D^+ consists of a single set of statements, and D^- is not considered (i.e., $D^- = \{\Box\}$).

The hypothesis space of RDT is a subset of the space of generalized Horn clauses determined by a user-specified declarative bias (rule schemata). As specified by the ILP_D definition, RDT returns all most general elements of its hypothesis space that are confirmed on the data according to a user-settable criterion (see section 7). Since the hypothesis space that is searched consists of individual clauses, each solution returned is independent and can in fact be based on different assumptions (different completed models).

Instantiation of $\models_D$. Since RDT is considering single datasets only, we are only concerned with $\models_D$ on single datasets. In RDT, the assumptions that are allowed to complete the model are limited in two ways. First of all, RDT will only make assumptions about the conclusion predicate, never about premise predicates (unless they occur in the conclusion). This corresponds to assuming that objects are completely described except for the information that is to be inferred.

Second, and more importantly, the amount of assumptions that can be made is limited in RDT through a user-settable *confirmation criterion* for hypotheses that is based on six elementary criteria. Assume we are considering a hypothesis $H = P \rightarrow C$ where P is a conjunction of premises, and C is the conclusion literal. RDT then provides the following numbers (where $\Gamma := B \cup D^+$ and $\overline{C} := D$ if $C = \neg D$ and $\overline{C} := \neg C$ otherwise):

- $pos(H) :=| \{\sigma \mid P\sigma \in \Gamma \text{ and } C\sigma \in \Gamma\} |$ (positive instances)
- $neg(H) :=| \{\sigma \mid P\sigma \in \Gamma \text{ and } \overline{C}\sigma \in \Gamma\} |$ (negative instances)
- $pred(H) :=| \{\sigma \mid P\sigma \in \Gamma \text{ and } C\sigma \notin \Gamma \text{ and } \overline{C}\sigma \notin \Gamma\} |$ (predicted instances)
- $total(H) :=| \{\sigma \mid P\sigma \in \Gamma\} | (= pos(H) + neg(H) + pred(H))$ (total instances)
- $concl(H) :=| \{\theta \mid C\theta \in \Gamma\} |$ (conclusion instances)
- $unc(H) :=| \{\theta \mid C\theta \in \Gamma \text{ and } \nexists \sigma = \theta\rho : P\sigma \in \Gamma\} |$ (uncovered instances)

Of these six criteria, the third one (*pred*) is especially interesting, as it directly measures the number of assumptions that are necessary to make a hypothesis true. In combination with *pos* or *total*, we can use this figure to give us a percentage of assumptions in relation to the number of entries which were already present in the knowledge. In our example, the above figures for h_1 and h_2 on $d_1^+ \cup d_2^+ \cup d_3^+$ would be[2]:

h_1 : *pos*: 4 *neg*: 0 *total*: 6 *pred*: 2 *unc*: 0
h_2 : *pos*: 8 *neg*: 0 *total*: 36 *pred*: 28 *unc*: 0

so we could use the condition *pred*/*total*<0.4 (inductive leap less than 40%) to accept h_1 while rejecting h_2.

Using the criterion *neg*, RDT in fact allows the user to soften the requirements of the ILP description learning task a little bit, e.g. by allowing a certain number or percentage of exception to a descriptive hypothesis. Thus if d^- were added to the knowledge base, we would get

h_1 : *pos*: 4 *neg*: 2 *total*: 8 *pred*: 2 *unc*: 0 h_2 : *pos*: 8 *neg*: 4 *total*: 64 *pred*: 52 *unc*: 0

so if we wanted to tolerate these exceptions, we could use *neg*/*total*<0.3 as a criterion.

Interestingly, just by using these elementary criteria, RDT can also be made to behave as if in the nonmonotonic setting, since all we need to do is to interpret anything that is predicted as a negative instance. So if we state $neg + pred = 0$, this forces RDT not to make any assumptions, as required by the nonmonotonic setting. In general, any confirmation criterion of the general setting can be changed into a confirmation criterion for the nonmonotonic setting simply by using $neg + pred$ whereever *neg* was used before.

[2]Note that in RDT, all of the figures except concl and unc are computed with respect to premise instances.

7.2 CLAUDIEN

Learning task. CLAUDIEN ("CLAusal DIscovery ENgine", [RB93]) generates a hypothesis in the form of a full clausal theory, i.e. in a larger hypothesis space than RDT. The form that clauses in the hypothesis may take is specified through a declarative bias facility/language that combines rule schemata [KW92] and clause sets [BG94]. As RDT, CLAUDIEN also looks for regularities in a single dataset D. The dataset D has the form of a definite clausal theory and has thus a unique minimal Herbrand model. A closed-world assumption is employed (nonmonotonic ILP), assuming that all facts that are not true in $\mathcal{M}^+(D)$ are false.

Instantiation of $\models_D$. CLAUDIEN discovers clauses that are (almost) true in $\mathcal{M}^+(D)$. The number $P(c)$ of substitutions that make the body and the head of clause c true and the number $N(c)$ of substitutions that make the body of c true and the head of c false are used in deciding which clauses to report. These criteria thus correspond directly to the pos and neg criteria of RDT. A minimum may be set on $B(c) = P(c) + N(c)$, as well as on $A(c) = P(c)/B(c)$ (intuitively, B corresponds to coverage and A to accuracy). Only clauses that are above the thresolds are reported. For clauses with empty heads, only the threshold on $B(c)$ applies which in this case is the number of violations of the clause.

7.3 ICDT

Learning task. Just as CLAUDIEN, ICDT ("Integrity Constraint Discovery Tool", [Eng94]) searches the space of full clauses (where literals in both body and head may be negated as well). As a successor RDT, however, it employs RDT's simpler declarative bias language (schemata extended to general clauses) and search strategy. Like RDT and unlike CLAUDIEN, ICDT is using the general open-world instantiation of the description learning task.

Instantiation of $\models_D$. ICDT also makes assumptions only about the conclusion predicates of the clause to be learned. On hypotheses with non-empty conclusions, ICDT [Eng94] uses the same basic confirmation criteria that RDT is using. For headless clauses, however, as pointed out for CLAUDIEN, such criteria cannot be applied so easily. Since any instance of the premises is a negative instance, *pos*, *pred* are always 0 (so are *concl* and *unc*), and $neg = total$, so it is impossible to construct sensible expressions that would give a degree of confirmation.

To remedy this situation, ICDT introduces a new elementary criterion called *nap* (not applicable) that is based on the observation that a hypothesis $l_1 \& \cdots \& l_n \rightarrow \Box$ can be transformed into the set of clauses

$$l_1 \rightarrow \ \backslash + l_2 \vee \cdots \vee \ \backslash + l_n$$
$$l_2 \rightarrow \ \backslash + l_1 \vee \ \backslash + l_3 \vee \cdots \vee \ \backslash + l_n$$

$\vdots$

$l_n \rightarrow \ \backslash + l_1 \vee \cdots \vee \backslash + l_{n-1}$

A "positive" instance of a headless clause is thus one that makes one body literal true while leaving at least one other body literal unknown. Formally the definition in ICDT is (for $H = l_1 \& \ \cdots \& \ l_n \rightarrow \square$):

- $nap(H) :=\mid \{\sigma \mid \exists i : l_i\sigma \in \Gamma \text{ and } \exists j \neq i : \nexists\theta : l_j\sigma\theta \in \Gamma\} \mid$

In our example, we would find that for the hypothesis

h_6: father(X,Y) & mother(Z,Y) $\rightarrow \square$

$nap(h_6)$= 6.

7.4 ICL

Learning task. ICL ("Inductive Constraint Logic", [DRVL95]) is a successor of CLAUDIEN, inheriting thus its hypothesis language and declarative bias facilities. ICL also inherits the closed-world assumption from CLAUDIEN. The principal difference is that ICL can use several datasets (examples). It employs the local definition of the description relationship. The datasets are assumed to be sets of ground facts. A definite clausal theory B can be provided as common background knowledge for all datasets.

Instantiation of $\models_D$. ICL looks for clauses that are true for (almost) all positive examples (interesting datasets) and false for (almost) all negative examples (uninteresting datasets). Its confirmation measures are based on $P(c)$, the number of positive, and $N(c)$, the number of negative examples (datasets) for which a clause c is true. The accuracy of a clause c is defined as $A(c) = P(c)/(P(c) + N(c))$. As a simple criterion, a threshold on the accuracy of clauses could be emposed. ICL, however, employs a more complicated statistical measure, known under the name of significance [DRVL95].

8 Future work: Towards a model-level characterization of degree of confirmation

What we have seen in the preceding section is that the implemented systems that address the ILP description problem have found some relatively flexible solutions to allow the user to accept certain hypotheses while rejecting other based on their "degree of confirmation". However, on closely inspecting the measures of RDT, ICDT and CLAUDIEN, they are based on counting substitutions with respect to the hypothesis, i.e., they need to make assumptions about the particular form of the hypothesis. Since the description problem itself is defined entirely on the model

level, and thus independent of a particular representation language, it would be nice to have a notion of degree of confirmation also on the model level.

At first sight, the solution chosen in ICL, where we have several sets of data, seems to offer a solution. ICL employs the closed-world assumption, but we can easily generalize the idea to the general open world description problem and define elementary criteria from which more complicated criteria could be constructed. As usual, let D be a set of data sets, B background knowledge, and H the hypothesis.

- $pos(H) :=| \{d \in D \mid B \cup d \models H\} |$
- $neg(H) :=| \{d \in D \mid B \cup d \cup H \models \square\} |$
- $pred(H) :=| \{d \in D \mid B \cup d \not\models HB \cup d \cup H \not\models \square\} |$
- $total(H) :=| \{d \in D\} | = pos(H) + neg(H) + pred(H)$

In our example (now taking d_1^+, d_2^+, and d_3^+ separately), we obtain:

h_1 : *pos*: 2 *neg*: 0 *total*: 3 *pred*: 1
h_2 : *pos*: 2 *neg*: 0 *total*: 3 *pred*: 1

or, if we also add d^- to the list of theories,

h_1 : *pos*: 2 *neg*: 1 *total*: 4 *pred*: 1
h_2 : *pos*: 2 *neg*: 1 *total*: 4 *pred*: 1

This shows that this definition of degree of confirmation actually is quite successful in estimating the "local" properties of hypotheses, but that it cannot distinguish between h_1 (a "local" hypothesis) and h_2 (a "global" hypothesis). In order to give a degree of confirmation of h_2, we necessarily must look at the "global" assumptions that are necessary to make the hypothesis true, and this, in effect, means that the distinction between the individual d_is cannot help us. We thus need (and unfortunately in this paper cannot offer yet) a model-based view of degree of confirmation of a hypothesis within a single theory.

To conclude, note that with the above definitions (just as with their substitution-based variants in RDT and ICDT) we can gradually change the systems behavior from the general descriptive setting to a nonmonotonic descriptive setting simply by limiting the size of *pred*. In the extreme, we treat *pred* just like *neg*, meaning every assumption is equivalent to a negative instance, resulting in exactly the nonmonotonic view. This again illustrates that the nonmonotonic semantics is a special case of the general description task.

9 Conclusion

In this paper, we have examined a data mining learning task in ILP that we have the ILP description learning problem. The ILP description problem is the learning task

of finding, within a given hypothesis space, a set of hypothesis that are descriptive of (confirmed on) the data. The ILP description learning problem is a generalization of the non-monotonic semantics view of ILP to the general open world (partial models) setting. As we have seen, the most important issue to be addressed in this general case is the definition of when a hypothesis describes a set/several sets of data. For several datasets, three possible definitions of the description relationship can sensibly employed. Based on these definitions, we showed that the description problem and the standard ILP prediction learning task cannot be reduced to each other: the prediction problem poses stronger requirements on hypotheses with respect to positive examples (entailment) whereas the description problem poses stronger requirements on the negative examples/uninteresting datasets (non-description).

For the basic description relationship on individual datasets, we saw that we need to restricts the kinds of models that need to be assumed to make a hypothesis true, ideally by computing some kind of degree of confirmation. Even though the implemented systems that address variants of the description learning task all have certain elementary criteria upon which such a degree of confirmation can be based, these criteria either are based on counting substitutions, and thus depend on the form of the hypothesis. The general model-level view based on counting the individual datasets on which a hypothesis is confirmed still leaves us with the problem of how to measure confirmation in the individual datasets and for "global" hypotheses. A central goal of future work should therefore be to try to find a model-level characterization of degree of confirmation on individual datasets, somehow measuring "how much" the partial model needed to be extended to make hypothesis true. For finite models, it might be possible to compute such figures directly, for infinite models, some sort of compression measure might be useful.

References

[BG94] Francesco Bergadano and Daniele Gunetti. Learning clauses by tracing derivations. In Stefan Wrobel, editor, *Proc. Fourth Int. Workshop on Inductive Logic Programming (ILP-94)*, pages 11 – 29, Schloß Birlinghoven, 53754 Sankt Augustin, Germany, 1994. GMD (German Natl. Research Center for Computer Science). GMD-Studien Nr. 237.

[DRB93] L. De Raedt and M. Bruynooghe. A theory of clausal discovery. In R. Bajcsy, editor, *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, pages 1058–1063. Morgan Kaufmann, 1993.

[DRD94] L. De Raedt and S. Džeroski. First order jk-clausal theories are pac-learnable. *Artificial Intelligence*, 70:375–392, 1994.

[DRVL95] L. De Raedt and W. Van Laer. Inductive constraint logic. Technical report, Departement Computerwettenschapen, Katholieke Universiteit Leuven, Leuven, Belgium, 1995.

[Dž95] S. Džeroski. *Numerical Constraints and Learnability in Inductive Logic Programming.* PhD thesis, Faculty of Electrical Engineering and Computer Science, University of Ljubljana, Ljubljana, Slovenia, 1995.

[Eng94] Roman Englert. Repräsentation, prüfen und lernen von integritätsbedingungen im mobal-system. Arbeitspapiere der gmd, GMD, St. Augustin, 1994. M.S. Thesis.

[Fla92] P.A. Flach. A framework for inductive logic programming. In S. Muggleton, editor, *Inductive Logic Programming*, pages 193–212. Academic Press, 1992.

[Hel89] Nicolas Helft. Induction as nonmonotonic inference. In *Proceedings of the 1st International Conference on Knowledge Representation and Reasoning*, pages 149 – 156, San Mateo, CA, 1989. Morgan Kaufman.

[KD94] Jörg-Uwe Kietz and Saso Dzeroski. Inductive logic programming and learnability. *SIGART Bulletin*, 5(1):22 – 32, 1994.

[KW92] Jörg-Uwe Kietz and Stefan Wrobel. Controlling the complexity of learning in logic through syntactic and task-oriented models. In Stephen Muggleton, editor, *Inductive Logic Programming*, chapter 16, pages 335 – 359. Academic Press, London, 1992. Presented at the Int. Workshop on Inductive Logic Programming, 1991. Also available as Arbeitspapiere der GMD No. 503.

[LD94] N. Lavrač and S. Džeroski. *Inductive Logic Programming: Techniques and Applications.* Ellis Horwood, 1994.

[Llo87] J.W. Lloyd. *Foundations of Logic Programming.* Springer Verlag, Berlin, New York, 2nd edition, 1987.

[MDR94] Stephen Muggleton and Luc De Raedt. Inductive logic programming: Theory and methods. *Journal of Logic Programming*, 19/20:629 – 679, 1994.

[Mug91] S. Muggleton. Inductive logic programming. *New Generation Computing*, 8(4):295–318, 1991.

[RB93] L. De Raedt and M. Bruynooghe. A theory of clausal discovery. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence.* Morgan Kaufmann, 1993.