Probabilistic Abductive Logic Programming: a Joint Approach of Logic and Probability

F. Rotella[1] and S. Ferilli[1,2]

[1] Dipartimento di Informatica – Università di Bari
{fulvio.rotella, stefano.ferilli}@uniba.it
[2] Centro Interdipartimentale per la Logica e sue Applicazioni – Università di Bari

1 Introduction and Background

In *Artificial Intelligence* (AI) two approaches have been attempted so far: numerical/statistical on one hand, and relational on the other. Statistical methods are not able to fully seize the complex network of relationships, often hidden, between events, objects or combinations thereof. This issue has been faced by exploiting First-Order Logic (FOL) for representing the world. But this setting holds when data are certain and complete and since this is far from being always true, there is the need for handling incompleteness and noise. Thus logical and statistical learning have been combined in the way that the former provides the representation language and reasoning strategies, and the latter enforces robustness. This gave origin to a new research area known as *Probabilistic Inductive Logic Programming* [7] (PILP) or *Statistical Relational Learning* [2] (SRL). Moreover working in very complex contexts often requires deductive reasoning to be supported by a variety of techniques that cope with the lack or incompleteness of the observations. Abductive inference can tackle incompleteness in the data by allowing to guess information that has not been explicitly observed [3]. Since there are so many explanations for such guesses, there is the need of selecting the most plausible one. While humans are able to discriminate such explanations, embedding this capability in machines is not trivial.

In the literature abductive reasoning has been performed by means of two main approaches: uncertainty has been faced by bayesian probabilistic graphical models and incompleteness by means of the classical approach based on pure logic. Nowadays many works combined logical abduction and statistical inference, which allows to rank all possible explanations and choose the best one. Mainly two approaches have merged directed and undirected graphical models with logic. The former [8] exploits Bayesian Logic Programs (BLPs) as a representation language for abductive reasoning; the latter [5] exploits Markov Logic Networks (MLN). It's worth noting that like MLNs, most SRL formalisms use deduction for logical inference, and hence they cannot be used effectively for abductive reasoning. Another approach [1] performs abduction using Stochastic Logic Programs [6] (SLP) by considering a number of *possible worlds*. Abductive reasoning is carried out by reversing the deductive flow of proof and collecting the probabilities associated to each clause. Although this approach is

probabilistically consistent with the SLP language, abduction through reversion of deduction is quite hazardous because abductive reasoning by means of deduction without constraints may lead to wrong conclusions. A more extensive comparison to other approaches has been presented in [9].

Our initial work has been focused on two perspectives: the generation of multiple (and minimal) explanations consistent with a given background knowledge, and the definition of a strategy to prioritize different explanations using their chance of being true. Now, in this paper, a new perspective is presented underlining the differences in the way of which such explanations are produced and accordingly how probabilities are associated to them.

2 Preliminaries

Our approach is based on Abductive Logic Programming [3] (ALP), a high-level knowledge representation framework that allows to solve problems declaratively based on abductive reasoning. It extends Logic Programming by allowing some predicates, called abducible predicates, to be incompletely defined. Problem solving is performed by deriving hypotheses on these abducible predicates (abductive hypotheses) as solutions of the problems to be solved.

Definition 1 (Abductive Logic Program). *An abductive logic program is made up of a triple $< P, A, IC >$, where:*

- *P is a standard logic program;*
- *A (Abducibles) is a set of predicate names;*
- *IC (Integrity Constraints or domain-specific properties) is a set of first order formulae that must be satisfied by the abductive hypotheses.*

These three components are used to define *abductive explanations*.

Definition 2 (Abductive Explanation). *Given an abductive theory $T =< P, A, IC >$ and a formula G, an* abductive explanation Δ *for G is a set of ground atoms of predicates in A s.t. $P \cup \Delta \models G$ (Δ explains G) and $P \cup \Delta \models IC$ (Δ is consistent). When it exists, T abductively entails G, in symbols $T \models_A G$.*

Consider a clause C: $h(t_1, .., t_n) :\text{-} l_1, ..l_{n'}$, where h is the unique literal of the head, n is its arity, and the l_i's, with $i = \{1, ..., n'\}$, are the literals in the body. In the following we will denote by $\overline{l_i}$ the negative literal $\neg l_i$. Let us consider Example 1, reporting a logic program P for concept $printable(X)$ (that describes the features that a document must own in order to be printed by a particular printer) and the set of abducibles A.

Example 1 (Theory for paper domain).   *Example 2 (Constraints).*

$c_1 : printable(X) \leftarrow a4(X), \overline{text(X)}$ $ic_1 = \langle 0.8, [table, text, image], or \rangle$

A = {image, text, b&w, printable, table, a4, a5, a3} $ic_2 = \langle 1.0, [a3, a4, a5], xor \rangle$

A proof procedure for abductive logic programs was presented in [4]. It interleaves phases of *abductive* and *consistency derivations*: an *abductive derivation*

is the standard Logic Programming derivation extended in order to consider abducibles. When an abducible literal δ has to be proved, it is added to the current set of assumptions (if not already included). Since the addition of δ must not violate any integrity constraint, a *consistency derivation* starts to check that all integrity constraints containing δ fail. In the *consistency derivation* an *abductive derivation* is used to solve each goal. This might cause an extension of the current set of assumptions. The procedure returns the minimal abductive explanation set if any, otherwise it fails.

3 The proposal

In order to motivate our approach we can assume that to abduce a fact, there is the need for checking if there are constraints that prevent such an abduction. The constraints can be either universally valid laws (such as temporal or physical ones), or domain-specific restrictions. Constraints verification can involve two kinds of hypotheses: those that can be deductively proved and those that have to be abduced. The former can be surely assumed true, conversely the latter might involve other abductions and thus there is the need of evaluating all possible situations before assuming the best one. In this view each abductive explanation can be seen as a *possible world*, since each time one assumes something he conjectures the existence of that situation in a specific world. Some abductions might be very unlikely in some worlds, but most likely in other ones. In this context we will refer to a *possible world* as the sets of the assumptions (i.e. hypothesized literals) and of the involved constraints that *must* hold in a particular *abductive explanation*.

We have also introduced new kinds of constraints in addition to typical ones consisting of a (non-probabilistic) logical *nand* of conditions. Specifically, we consider constraints that may express in a more suitable and understandable way some situations, by using also the logical operators *or* and *xor*. Moreover due to noise and uncertainty in the real world, we have smoothed each constraint with a probability that reflects the degree of the personal belief in the likelihood of the whole constraint. In this perspective we can extend the classical Definition 1 of abductive logic programs by replacing the concept of *Integrity Constraints* with that of *Typed Probabilistic Constraints*.

Definition 3 (Extended Abductive Logic Program). *An extended abductive logic program ET is made up of a triple $\langle P, A, TPIC \rangle$, where P is a standard logic program, A (Abducibles) is a set of predicate names and $TPIC$ (Typed Probabilistic Constraints) is a set of probabilistic constraints whose elements are triples of the form $\langle Pr, S, T \rangle$ where Pr is a probability and expresses its reliability, or our confidence on it, T is the* type *of constraint and represents the kind of denial, and S is the set of literals of the constraint that must be satisfied by the abductive hypotheses. The way in which the constraints must be satisfied is defined by its type.*

Let us consider Example 2: ic_1 states that a document can be composed either of tables, images or text and that our personal belief in the likelihood of

this constraint is 0.8. In other words we believe that a document is composed 80% of times of at least tables, images or text, and that only 20% of times a document does not have at least one of such elements. The latter constraint ic_2 states that a document must be at least of one of the three formats and moreover that this situation is always true.

In our previous work we focused on the study of what happens in such *possible worlds* in which all the activated constraints (i.e. constraints involved in the proof procedure) must hold, and thus we discarded all the other worlds in which such constraints do not hold. Conversely in this paper we take into account all the *possible worlds* in which all activated constraints might hold or not. According to this new perspective we have further introduced in this paper a formal definition for our *probabilistic abductive explanation.*

Our new approach can be described from two sides: the logical proof procedure and the computation of probabilities.

3.1 Logical perspective

Our procedure consists of an extended version of the classical approach [3, 4]. While it resolved the goal by looking for one minimal abductive explanation set, our approach is to generate different abductive explanations and then evaluate them all. It can be achieved by changing each assumption that may constrain the subsequent explanations. Thus each time the procedure assumes something two *possible worlds* can be considered: one where the assumption holds and another where it does not. This idea has been extended also to the *probabilistic constraints.* In fact our previous perspective was more strict since we kept a *possible world* iff it falsified all the constraints involved in the proof (activated constraints). Conversely such an assumption can be relaxed, considering *possible worlds* consistent with a subset of the constraints activated by the proof procedure. In such a perspective the overall view is analogous to the exploration of a tree in which each interior node corresponds to a decision (an assumption), as for example changing the truth value of a literal, an edge represents the consequences of that decision and a leaf is the conclusion of the abductive reasoning in a particular *possibile world.* It is worth noting that in this way we have to consider a notion of partial consistency since in each explanation the involved constraints might be either falsified or not.

With such objectives we forced the classical backtracking procedure in order to: assume both values (positive and negative) of a literal, choose different predicate definitions (i.e. choosing different definitions, other literals will be taken into account and thus other constraints are involved) and consider the constraints involved in each abductive explanation as falsified or not.

3.2 Probabilistic perspective

Given all explanations, the issue of selecting the best one arises. The abduction probability of each ground literal through *the possible worlds* can be obtained

considering both the chance of it being true in the real world and all sets of assumptions made during the abductive proof in each *possible world*.

Let's introduce some notation: $\Delta = \{P_1 : (\Delta_1, Ic_1), ..., P_T : (\Delta_T, Ic_T)\}$ is the set of the T consistent *possible worlds* that can be assumed for proving a goal G. Each (Δ_i, Ic_i) is a pair of sets: $\Delta_i = \{\delta_{i1}, ..., \delta_{iJ_i}\}$ contains the ground literals δ_{ij} abduced in a single abductive proof, and $Ic_i = \{ic_{i1}, ..., ic_{iK_i}\}$ is the set of the constraints ic_{ik} involved in the explanation Δ_i. Both Δ_i and Ic_i may be empty. Moreover, we have used the following symbols in our equations: $n(\delta_{ij})$ is the number of true grounding of the predicate used in literal δ_{ij}, $n(cons)$ is total number of constants encountered in the world, $a(\delta_{ij})$ is the arity of literal δ_{ij} and $P(ic_{ik})$ is the probability of the k-th constraint in Ic_i. The chance of being true of a ground literal δ_{ij} can be defined as in Equation 1. Then the unnormalized probability of the abductive explanation can be assessed by Equation 2 and the probability of the abductive explanation normalized over all T consistent worlds can be computed as in Equation 3:

$$P(\delta) = \frac{n(\delta)}{\frac{n(cons)!}{(n(cons)-a(\delta))!}} \quad (1) \qquad P'_{(\Delta_i, Ic_i)} = \prod_{j=1}^{J_i} P(\delta_{ij}) * \prod_{k=1}^{K_i} P(ic_{ik}) \quad (2) \qquad P_{(\Delta_i, Ic_i)} = \frac{P'_{(\Delta_i, Ic_i)}}{\sum_{t=1}^{T} P'_{(\Delta_t, Ic_t)}} \quad (3)$$

Equation 1 expresses the ratio between true and possible groundings of literal δ. The intuition behind this equation can be expressed with a simple example: given a feature $f(\cdot)$ and an item obj that does not have such a feature, if we want to assess the probability that obj owns $f(\cdot)$ (i.e. $P(f(obj))$), we should consider how often we found items for which $f(\cdot)$ holds over all items for which it might hold in real world.

The probability of $\overline{\delta}$ is equal to $1-P(\delta)$ as well as $P(\overline{ic})$ is equal to $1-P(ic)$. It is worth noting that we are considering the Object Identity [10] assumption and so within a clause, terms (even variables) denoted with different symbols must be distinct. Thus only literal groundings without constants repetitions are allowed in Equation 1. The first part of the Equation 2 encloses the probability that all abduced literals are true in that particular *world*. The second part expresses the reliability of the constraints involved in the i-th abductive explanation.

In our previous work some worlds are ruled out due to inconsistency with all integrity constraints and so in the computation of the probabilities we do not consider such situations. Conversely, in this paper we have taken into account also the *possible consistent world* in which some activated constraints might not hold. Now, we can state a definition for our *probabilistic abductive explanation*:

Definition 4 (Probabilistic Abductive Explanation). *Given an extended abductive theory $ET = \langle P, A, TPIC \rangle$ and a formula G, a probabilistic abductive explanation $\langle \Delta^*, IC^* \rangle$ for G is a pair of sets, where Δ^* is a set of ground literals l of predicates in A, IC^* with $IC^* \subseteq TPIC$ is the set of constraints involved, whose $IC_H \subseteq IC^*$ is the subset of the ones which hold in such abductive explanation Δ^*, s.t. $P \cup \Delta^* \models G$ (Δ^* explains G) and $P \cup \Delta^* \models IC_H$ (Δ^* is consistent). When it exists at least a pair $\langle \Delta^*, IC^* \rangle$ whose $P_{(\Delta^*, IC^*)} > 0$ and $P_{(\Delta^*, IC^*)}$ obtained as in Equation 3, ET abductively entails G under at least a possible world (i.e. $\langle \Delta^*, IC^* \rangle$), in symbols $ET \models_{\langle \Delta^*, IC^* \rangle} G$.*

4   Conclusion

Reasoning in very complex contexts often requires pure deductive reasoning to be supported by a variety of techniques that can cope with incomplete and uncertain data. Abductive inference allows to guess information that has not been explicitly observed. Since there are many explanations for such guesses, there is the need for assigning a probability to them in order to choose the best one. Our initial work was focused on two perspectives: the generation of multiple (and minimal) explanations consistent with a given background knowledge, and the definition of a strategy to prioritize different explanations using their chance of being true. In this paper a new perspective is presented underlining the differences in the way of which such explanations are produced and accordingly how the probability are related to. The new approach was tested on some toy experiments motivating further research. Since our old procedure has been tested on some standard datasets, we will exploit this extended procedure in the same experimental setting in order to compare both approaches. Further studies will be focused on learning the probabilistic constraints and on exploiting our approach in other tasks such as natural language understanding and plan recognition.

References

[1] Andreas Arvanitis, Stephen H. Muggleton, Jianzhong Chen, and Hiroaki Watanabe. Abduction with stochastic logic programs based on a possible worlds semantics. In *In Short Paper Proc. of 16th ILP*, 2006.

[2] Lise Carol Getoor. *Learning statistical models from relational data.* PhD thesis, Stanford, CA, USA, 2002. AAI3038093.

[3] A. C. Kakas, R. A. Kowalski, and F. Toni. Abductive logic programming, 1993.

[4] Antonis C. Kakas and Fabrizio Riguzzi. Abductive concept learning. *New Generation Comput.*, 18(3):243–294, 2000.

[5] Rohit J. Kate and Raymond J. Mooney. Probabilistic abduction using markov logic networks. In *Proceedings of the IJCAI-09 Workshop PAIR*, Pasadena, CA, July 2009.

[6] S. Muggleton. Stochastic Logic Programs. In L. De Raedt, editor, *Proceedings of the 5th ILP*. Dept. of CS, Katholieke Universiteit Leuven, 1995.

[7] Luc De Raedt and Kristian Kersting. Probabilistic inductive logic programming. In *ALT*, pages 19–36, 2004.

[8] Sindhu V. Raghavan. Bayesian abductive logic programs: A probabilistic logic for abductive reasoning. In Toby Walsh, editor, *IJCAI*, pages 2840–2841, 2011.

[9] Fulvio Rotella and Stefano Ferilli. Probabilistic abductive logic programming using possible worlds. In *28th Italian Conference on Computational Logic (CILC-2013)*, 2013.

[10] G. Semeraro, F. Esposito, D. Malerba, N. Fanizzi, and S. Ferilli. A logic framework for the incremental inductive synthesis of datalog theories. In Norbert E. Fuchs, editor, *LOPSTR*, volume 1463 of *LNCS*, pages 300–321. Springer, 1997.