Probabilistic Similarity Logic

Matthias Broecheler, Lise Getoor MATTHIAS@CS.UMD.EDU, GETOOR@CS.UMD.EDU
University of Maryland, A.V. Williams Bldg, College Park, MD 20742 USA

Abstract

Many interesting research problems, such as ontology alignment and collective classification, require probabilistic and collective inference over imprecise evidence. Existing approaches are typically ad-hoc and problem-specific, requiring significant effort to devise and provide poor generalizability. In this paper, we introduce *probabilistic similarity logic (PSL)*, a simple, yet powerful language for describing problems which require probabilistic reasoning about similarity where, in addition to reasoning probabilistically, we want to capture both logical constraints and imprecision. We prove that PSL inference is polynomial and outline a wide range of application areas for PSL.

1. Introduction

Many problems addressed by statistical relational learning (SRL), such as ontology alignment and collective classification, exhibit four important characteristics:

Collective Decision: The overall decision problem is composed of a (large) number of individual decisions which depend on one another in complex ways.

Locality of Evidence: In solving the decision problem, only local evidence is considered, which means that each individual decision depends only on a small number of other decisions which are factored into the decision process.

Imprecision of Decision: Individual decisions can be vague and imprecise, meaning that one cannot establish a decision variable to be either true or false.

Probabilistic Evidence: The evidence used in deciding individual decision variables is of a probabilistic nature.

Presented at *ILP-MLG-SRL*, Leuven, Belgium, 2009.

We call the class of decision problems which posses the above four characteristics CLIP, for collective, local, imprecise, and probabilistic.
Locality facilitates decomposition of the problem and probabilities are a popular model for uncertainty. Both concepts are frequently employed in SRL approaches. We used the term "collective" instead of "relational" to highlight the fact that dependencies are frequently symmetric therefore rendering directed models unapplicable. The novel concept of **imprecision** in this context is particularly important when reasoning about similarity where boolean truth values are too constraining. Imprecision and uncertainty are distinct concepts.

CLIP contains a number of interesting and widely studied problems. Yet, most of these problems have been studied in isolation and ad-hoc approaches have been proposed to their solution. We propose a general-purpose framework, called *probabilistic similarity logic* (PSL), as a first attempt at a unifying framework for reasoning about CLIP problems. PSL is based on Markov Random Fields and t-norm Fuzzy Logics thereby combining reasoning about uncertainty and imprecision in one framework. Furthermore, PSL is the conclusion of a careful analysis of CLIP problems and provides features that cater toward the particular requirements of this class of problems. Benefiting from the generality of a logical language, PSL can describe a great variety of problems; we demonstrate its use on the two initially mentioned research topics. In addition, we identify a fragment of PSL which allows for an optimal, polynomial time inference algorithm and verify its efficiency in practice.

Existing general frameworks for probabilistic reasoning, such as Markov Logic (Richardson & Domingos, 2006) and Bayesian Logic Programs (Kersting & Raedt, 2001), lack the expressiveness to describe all four aspects of CLIP problems. In addition, inference in such frameworks can be NP-hard in the worst case, very time consuming in practice, and often relies on probabilistic approximation algorithms. On the other hand, individual ad-hoc approaches require significant implementation effort, are idiosyncratic, and do not generalize well to similar problems.

2. Similarity Logics

This section defines the basic syntax and semantics of *probabilistic similarity logic* (PSL) as used in *probabilistic similarity logic programs* (PSLP). We assume that any problem of interest is logically describable by a finite set of *predicate symbols* $\mathcal{P}$ (each with an associated finite arity), a finite set of *constant symbols* $\mathcal{C}$, and an infinite set of *variable symbols* $\mathcal{V}$. A *term* is either a constant symbol or a variable symbol. Let $p \in \mathcal{P}$ be an n-ary predicate symbol and let $t_1, \ldots, t_n \in \mathcal{C} \cup \mathcal{V}$ be terms, then $p(t_1, \ldots, t_n)$ is an *atom*. If all $t_i \in \mathcal{C}$ are constant symbols, then $p(t_1, \ldots, t_n)$ is a *ground atom*.

Definition 1 (PSL-Rule) *Let $B_1, \ldots, B_n$, H be atoms and $\mu \in \mathbb{R} \cup \{\infty\}$ a real number or infinity, then*

$$H \overset{\mu}{\leftarrow} B_1 \wedge B_2 \wedge \ldots \wedge B_n$$

*is a PSL-Rule. μ is called the weight of a PSL-Rule. A PSL-Rule with finite weight is called a probabilistic **evidence rule**, and **certainty rule** if the weight is infinite.*

A variable substitution $\sigma : \mathcal{V} \rightarrow \mathcal{C}$ is used to ground non-grounded rules. For a given PSL-rule r, σr denotes a grounded instance whereby all free variable symbols v in r are replaced by $\sigma(v)$.

PSL is a real-valued logic which means that each ground atom can assume a truth value from the unit interval $[0, 1]$ instead of being limited to the boolean truth values *true* and *false*. Intuitively, the truth value 0 corresponds to *false* and 1 to *true* and any value $v \in (0, 1)$ represents a degree of truth between those two extrema. As we shall see in the following, real-valued *degrees of truth* allow us to model the imprecision inherent in CLIP problems.

Definition 2 (PSL-Interpretation) *A PSL-Interpretation (or truth evaluation) I is mapping $I : \mathcal{A}_G \rightarrow [0, 1]$ from the set of all ground atoms $\mathcal{A}_G$ to the unit interval.*

The semantics of PSL are based on those of Fuzzy Logics (Klir & Yuan, 1995),(Gottwald, 2001). As in t-norm fuzzy logics (Esteva & Godo, 2001), we derive the truth value of formulas from the truth values of its atoms using so called t-norm operators. For our purposes it suffices to think of a t-(co)norm $\circ$ as a "truth combination function" $\circ : [0, 1] \times [0, 1] \rightarrow [0, 1]$ t-(co)norms allow us to extend a PSL-Interpretation to entire formulas and rules.

Definition 3 (Semantics of Rules) *Let I be a PSL-Interpretation and let $\oplus$ be a t-conorm, then we define the truth value of a PSL-Rule as:*

$$I(H \overset{\mu}{\leftarrow} B_1 \wedge \ldots \wedge B_n) = (\oplus_i I(\neg B_i)) \oplus I(H)$$

The syntax and semantics of PSL presented thus far allow us to capture facts and the inherent imprecision of CLIP problem instances, as well as formalize our knowledge about the problem using rules. The uncertainty that underlies the class of problems we are interested in is modeled as a probability distribution over the set of interpretation. Since there is no single "correct" interpretation, we have to reason about the relative likelihood as prescribed by the weighted rules. The evidence rules $\mathcal{R}_E$ represent uncertain knowledge about the problem domain. The weight of a rule $r \in \mathcal{R}_E$ reflects the likelihood that a rule holds in the domain. A (relatively) high weight μ of evidence rule r means that it is very likely that each grounded instance σr has a high degree of truth.

Definition 4 (Probability of Interpretation) *Let $\mathcal{R}_E$ be a set of evidence rules and $\mathcal{R}_C$ be a set of certainty rules. We define the probability of any interpretation I w.r.t $\mathcal{R}_E, \mathcal{R}_C$ as:*

$$\mathbb{P}(I|\mathcal{R}_E, \mathcal{R}_C) = \begin{cases} \frac{1}{Z} e^{\sum_{r \in \mathcal{R}_E} \sum_{\sigma} \mu_r \times I(\sigma r)} & \text{if } \forall r \in \mathcal{R}_C, \sigma : I(\sigma r) = 1 \\ 0 & \text{otherwise.} \end{cases}$$

where $Z = \int_{I, \forall r \in \mathcal{R}_C, \sigma : I(\sigma r) = 1} e^{\sum_{r \in \mathcal{R}_E} \sum_{\sigma} \mu_r \times I(\sigma r)}$ is a normalization constant to ensure that $\mathbb{P}$ is a valid point probability distribution.

The grounded evidence rules induce a Markov Random Field (Kindermann & Snell, 1980) over the set of ground atoms constrained by the grounded certainty rules. The probabilistic model is therefore similar to that of Markov Logic (Richardson & Domingos, 2006) with the difference that PSL uses point probabilities over a high dimensional hypercube and differentiates between evidence and certainty rules.

To effectively address some of the characteristics frequently observed in CLIP problems, we extend PSL by some novel constructs.

In PSL we distinguish between three types of predicate symbols and correspondingly between three types of atoms: *basic atoms*, *composite atoms*, and *attribute atoms*. **Basic atoms** are just normal atoms and have no special meaning.

Composite atoms are atoms whose truth values depend on the truth value of other atoms. In collective decision problems, one frequently needs to consider a particular aggregate over a set of related decisions. Composite atoms are the means to define such aggregates in PSL.

As the name suggests, **attribute atoms** are restricted to attribute symbols, i.e., the constant terms of ground attribute atoms must correspond to attributes in the domain, such as strings and numbers. Constraining the signature of attribute predicates in this manner has the advantage of allowing users to define arbitrary truth evaluation functions for attribute atoms externally which makes it possible to easily incorporate sophisticated and well-studied attribute similarity measures, such as string similarity measures (Chandel et al., 2007), into the PSL framework.

The definition of a probability distribution over PSL-Interpretations given in the previous section considers grounded evidence rules to be independent. In most cases, this independence assumption is (approximately) true and allows us to factor the probability distribution into manageable parts. Yet, there are cases when it is necessary to account for the dependencies between rules. For this, we introduce **p-junctions**, which are special combiner operators similar to combining rules in Bayesian Logic Programs (Kersting & Raedt, 2001).

Definition 5 (p-junction) *Let* $H_1, \ldots, H_n$ *be atoms, then* $H_1 \Diamond H_2 \Diamond \ldots \Diamond H_n$ *is a* ***p-junction****. A* ***p-junction operator*** $\odot$ *is a function* $\odot : [0,1] \times [0,1] \rightarrow [0,1]$ *such that* $\otimes(x,y) \leq \odot(x,y) \leq \oplus(x,y), \forall x,y \in [0,1]$*. The semantics of a p-junction are then given by* $I(H_1 \Diamond \ldots \Diamond H_n) = \odot_i H_i$*.*

Definition 6 (PSL Program) *A probabilistic similarity logic program (PSLP) P is a tuple* $P = \langle \mathcal{R}_E, \mathcal{R}_C, KB \rangle$*, where* $\mathcal{R}_E$ *is a set of evidence rules,* $\mathcal{R}_C$ *is a set of certainty rules, and* $KB \langle \mathcal{A}_{KB}, I_{KB} \rangle$ *is a PSL-Knowledge Base of atoms* $\mathcal{A}_{KB}$ *and their truth values* I_{KB}*.*

Given a PSLP, one is naturally interested in the truth values of those atoms which are not included in the knowledge base, i.e. in the truth value of the non-evidence atoms which we do not know the truth value a priori. In the context of reasoning under uncertainty, this is the problem of finding the most likely truth value assignment.

Definition 7 (Most Probable Interpretation) *Given a PSLP* $P = \langle \mathcal{R}_E, \mathcal{R}_C, KB \rangle$*, the most probable interpretation (MPI)* I_{MPI} *is defined as:*

$$I_{MPI} = argmax_I \mathbb{P}(I \mid \mathcal{R}_E, \mathcal{R}_C, I = I_{KB} \text{ on } \mathcal{A}_{KB})$$

The MPI problem defined above essentially corresponds to finding the MAP state of the probability distribution $\mathbb{P}$. MPI-Inference in existing general frameworks is typically *NP*-hard. In contrast, we can show that finding the most probable interpretation of a PSLP is polynomial in the number of ground rules and atoms.

Theorem 1 *Under the assumptions that (1)* $\oplus$ *is the Lukasiewicz t-conorm:* $\oplus(x,y) = \min(x+y, 1)$*, (2)* $\odot$ *is a quadratic p-junction operator, and (3) the aggregator function for all composite atoms is linear in the truth values, MPI inference for a given PSLP can be reduced to a convex optimization problem which is solvable in time polynomial in the number of ground rules and atoms.*

Essentially, using real-valued logics yields a continuous optimization problem and the careful design of PSL with mild assumptions ensure convexity. PSL MPI inference can be transformed into a Second-Order Cone Program (SOCP) (Lobo et al., 1998), for which very efficient interior point methods have been developed (Nesterov & Nemirovsky, 1994) and shown to converge fast in practice. In fact, several high performing, commercial optimization toolboxes are available.

3. Applications and Future Work

Using ontology alignment and collective classification as examples, we demonstrated the ease with which such CLIP problems can be modeled and reasoned about in PSL while achieving performance that is comparable to existing, highly optimized ad-hoc approaches. Writing Prolog style rules is simple and intuitive, which highlights the utility of PSL as a general purpose, rapid prototyping framework for CLIP problems.

The expressivity and flexibility of PSL makes it applicable to a wide range of problems in computer vision, computational biology, database schema integration, natural language processing, social network analysis, and others which we hope to explore in future research to determine the scope and limitations of PSL.
In addition, we will investigate efficient learning of rule weights using Quadratic Programming.

Acknowledgements

The authors gratefully acknowledge funding support provided by the NSF through grant no. 0746930.

References

Chandel, A., Hassanzadeh, O., Koudas, N., Sadoghi, M., & Srivastava, D. (2007). Benchmarking declarative approximate selection predicates. *SIGMOD*. Beijing, China.

Esteva, F., & Godo, L. (2001). Monoidal t-norm based logic: towards a logic for left-continuous t-norms. *Fuzzy Sets and Systems*, *124*, 271–288.

Gottwald, S. (2001). A treatise on Many-Valued logics. *Studies in Logic and Computation*, *9*.

Kersting, K., & Raedt, L. D. (2001). *Bayesian logic programs* (Technical Report). Albert-Ludwigs University.

Kindermann, R., & Snell, J. L. (1980). *Markov random fields and their applications*. American Mathematical Society Providence, RI.

Klir, G. J., & Yuan, B. (1995). *Fuzzy sets and fuzzy logic: theory and applications*. Prentice Hall Upper Saddle River, NJ.

Lobo, M. S., Vandenberghe, L., Boyd, S., & Lebret, H. (1998). Applications of second-order cone programming. *Linear Algebra and Its Applications*, *284*, 193–228.

Nesterov, Y., & Nemirovsky, A. (1994). Interior point polynomial methods in convex programming. *Studies in applied mathematics*, *13*.

Richardson, M., & Domingos, P. (2006). Markov logic networks. *Machine Learning*, *62*, 107–136.