Symmetry Breaking for Relational Weighted Model Finding

Tim Kopp
Parag Singla
Henry Kautz

WL4AI
July 27, 2015
Outline

- **Weighted logics.**
- Symmetry-breaking in SAT.
- Symmetry-breaking for weighted logics.
Weighted Logics

- Logical AI techniques handle strong relational dependencies very efficiently.
- Statistical AI techniques are readily able to model uncertainty.
- *Statistical Relational Learning* (SRL) is a sub-field of AI that tries to accomplish both.
Weighted Logic Systems

- **Probabilistic Relational Models** — [Getoor et al., 2007]
- **Relational Dependency Networks** — [Neville and Jensen, 2007]
- **Bayesian Logic Programs** — [Kersting and De Raedt, 2007]
- **Probabilistic Soft Logic** — [Broecheler and Getoor, 2009]
- **BLOG** — [Milch et al., 2004]
- **Markov Logic Networks** — [Richardson and Domingos, 2006]
If $x$ is the weighted theory, and $y$ is the truth assignment, then the probability distribution is

$$P(y|x) = \frac{1}{Z(x)} \exp \left( \sum_i w_i n_i(x, y) \right)$$

Where $w_i$ is the weight of first order clause $i$, and $n_i$ is the number number of groundings of $i$ satisfied by $y$. 
The problem of *Most Probable Explanation in Markov Logic* (MPE) inference is to determine the most probable possible world, given the MLN and the evidence:

\[
\arg\max_y P(y|x) = \arg\max_y \frac{1}{Z(x)} \exp \left( \sum_i w_i n_i(x, y) \right) \\
= \arg\max_y \sum_i w_i n_i(x, y)
\]

This reduces to (partial, weighted) MaxSAT!
Outline

- Weighted logics
- **Symmetry-breaking in SAT.**
- Symmetry-breaking for weighted logics.
Symmetry Breaking

Exploit symmetries in a propositional CNF formula to reduce computational complexity of SAT-solving [Crawford et al., 1996]. Motivating example: Pigeon-hole problem.
Motivating Example: Pigeonhole Problem

- $P_{i,j}$: Pigeon $i$ is in hole $j$.
- At most one pigeon per hole:
  \[
  \bigwedge_{j=1}^n \neg P_{i_1,j} \lor \neg P_{i_2,j} \quad \forall i_1, i_2, i_1 \neq i_2
  \]
- Every pigeon in some hole:
  \[
  \bigvee_{j=1}^n P_{i,j} \quad \forall i
  \]

(As formulated in [Crawford, 1992])
Consider 10 pigeons, 9 holes.

- There are $2^{90}$ possible assignments.
- There are $10!$ search branches that reach the leaf before reaching UNSAT.
- Taking symmetry into account, there is one such search path.
- For the pigeonhole problem, we reduce the complexity from exponential to polynomial [Crawford, 1992].
A permutation of a finite set $S$ is a one-to-one mapping $\theta : S \to S$.

$Sym(S)$ denotes the set of permutations of $S$.

Let $s \in S$ and $\theta \in Sym(S)$. $s^\theta$ denotes $\theta(s)$, the image of $s$ under $\theta$. 

Consider a set of variables $L$, and its permutations $Sym(L)$.

- Let $\theta \in Sym(L)$ and $v \in L$. Then $(\neg v)^\theta \equiv \neg v^\theta$.
- Let $C = \bigvee_{i=1}^r v_i$ be a clause over $L$. Then $C^\theta \equiv \bigvee_{i=1}^r v_i^\theta$.
- Let $T = \bigwedge_{i=1}^s C_i$ be a theory over $L$. Then $T^\theta \equiv \bigwedge_{i=1}^s C_i^\theta$.
- Let $A$ be an assignment of $T$. Then $\theta A(v) \equiv A(v^\theta)$, and likewise $\theta A(T) \equiv A(T^\theta)$.
Symmetries in Propositional Logic

- $\theta \in \text{Sym}(L)$ is an automorphism, or symmetry of $L$ iff $T^\theta = T$.
- The set of symmetries is $\text{Aut}(L) \subseteq \text{Sym}(L)$.

Key idea:

$$A \in \mathcal{M}(T) \iff \theta A \in \mathcal{M}(T), \quad \theta \in \text{Aut}(T)$$
Symmetries in Practice

- We find symmetries through a reduction to graph isomorphism.
- $\text{Aut}(T)$ induces an equivalence relation (orbit) on the set of assignments:
  \[ A \equiv B \quad \text{iff} \quad B = \theta A \quad \theta \in \text{Aut}(T) \]
- To test satisfiability, it suffices to search only one assignment in each orbit.
- We create a theory $T'$ by adding symmetry-breaking predicates (SBPs) to $T$ such that:
  - $T$ and $T'$ are equisatisfiable.
  - $T'$ admits only one (or some) of the assignments in each orbit.
Consider a pigeonhole problem with 2 pigeons and 1 hole. The assignments:

\[ P_{1,1}, \neg P_{2,1} \]
\[ \neg P_{1,1}, P_{2,1} \]

are equivalent. We restrict which assignments are admitted by adding the following to the theory:

\[ P_{1,1} \Rightarrow P_{2,1} \]
Consider a pigeonhole problem with 3 pigeons and 2 holes. The assignments:

\[
P_{1,1}, \neg P_{2,1}, \neg P_{3,1}
\]

\[
\neg P_{1,1}, P_{2,1}, \neg P_{3,1}
\]

\[
\neg P_{1,1}, \neg P_{2,1}, P_{3,1}
\]

are equivalent. We **cannot** simply extend what we did for the previous care:

\[
P_{1,1} \Rightarrow P_{2,1} \Rightarrow P_{3,1}
\]

\[
P_{1,2} \Rightarrow P_{2,2} \Rightarrow P_{3,2}
\]
SBPs: Gating Condition

Solution: Only restrict the models when previous atoms have the same truth value.

\[
P_{1,1} \Rightarrow P_{2,1} \]

\[
(P_{1,1} \Leftrightarrow P_{2,1}) \Rightarrow P_{2,1} \Rightarrow P_{3,1}
\]

\[
P_{1,2} \Rightarrow P_{2,2} \]

\[
(P_{1,2} \Leftrightarrow P_{2,2}) \Rightarrow P_{2,2} \Rightarrow P_{3,2}
\]
- We impose an ordering on the variables $v_1, \ldots, v_n$.
- View an assignment on the variables as a binary number.
- We admit only the smallest model in any symmetry group.
SBPs: The Crawford formulation

For a theory over $L = \{v_1, \ldots, v_n\}$, we break symmetry $\theta \in Aut(L)$ by adding:

$$\left( \bigwedge_{j=1}^{i} v_j \Leftrightarrow v_j^\theta \right) \Rightarrow v_i \Rightarrow v_i^\theta, \quad \forall i$$

In general there are an exponential number of symmetries, making this intractable. In practice we only break a subset of symmetries. There are tools (e.g. Shatter [Aloul et al., 2003]) that build on this formulation.
Symmetry-Breaking for Inference

- Propositional SBPs preserve satisfiability.
- For MaxSAT, they create theories whose max model has the same number of satisfied clauses [Niepert, 2012].
- The same is true for partial weighted MaxSAT.
Outline

- Weighted logics
- Symmetry-breaking in SAT.
- Symmetry-breaking for weighted logics.
Goal: Leverage symmetries in MPE inference for weighted logics. Let:

- $\mathcal{C} = \{C_1, \ldots, C_k\}$ be the set of constants (terms).
- $T$ be a relational theory (as in Markov Logic), consisting of:
  - $E$ be the evidence, a set of hard ground literals.
  - $F$ be the set of weighted and hard first-order formulas.
Let $\theta$ be a permutation of the terms $C$.

If $a = P(C_i, \ldots, C_j)$ is an atom, then $a' = P(C_i^\theta, \ldots, C_j^\theta)$.

$\theta$ is a term symmetry iff $\theta(E) = E$.

Term symmetries can be induced from evidence by reduction to colored graph isomorphism.
Suppose we have evidence:

\[ P(A, B), \quad P(X, Y), \quad \neg Q(Y), \quad \neg Q(B) \]

Then \((A \ X)(B \ Y)\) is a term symmetry.
A term-equivalent symmetry is a set of term symmetries that induce equivalence classes on the terms. Given evidence:

\[ P(A), P(B), P(C), Q(A) \]

In the above example, the term-equivalent classes are \[ \{\{B, C\}, \{A\}\} \].
Let the *context* of a term be the sorted list of ground predicate positions it appears in.

\[ \neg P(A) \]
\[ Q(B, A, C) \]

In the above example, the context of \( A \) is \( \neg P(\ast), Q(B, \ast, C) \).

Two terms are term-equivalent iff they have the same context.

Efficient algorithm: Use a hash table to store and retrieve the context of equivalent terms.

Term-equivalent symmetry detection takes polynomial time.
Breaking Term-Equivalent Symmetries

Let:
- \( G_1, G_2, \ldots G_r \) be an ordering on the ground atoms.
- \( \theta \) be a symmetry that swaps two term-equivalent terms.
- The SBP to break \( \theta \) is:

\[
SBP(\theta) = \bigwedge_{1 \leq i \leq r} \left( \bigwedge_{1 \leq j \leq i} G_j \Leftrightarrow \theta(G_j) \right) \Rightarrow G_i \Rightarrow \theta(G_i)
\]

- To break all term-equivalent symmetries, add SBP for each pair of terms.
Suppose our theory is such that all of the predicates have at most one argument of each type. None of the arguments can interact with each other, so we need only a linear number of SBPs:

\[ \bigwedge_{l=1}^{m} SBP(\theta_l, l+1) \]
Term-equivalent SBPs preserve satisfiability.

For MPE, they create theories whose max model has the same weight.
Summary of Contributions

- Discovered term symmetries for relational theories.
- Discovered term-equivalent theories, and an algorithm for efficiently detecting them, independent of the theory.
- Constructed an SBP formulation for term-equivalent symmetries.
Future Work

- Implement and test against other systems.
  - Submission pending.
- Explore the relationship between our SBPs and existing lifted inference algorithms.
- Exploit term symmetries for marginal inference.
- Explore ways to exploit term and term equivalent symmetries during search.


