Sound and Efficient Inference with Probabilistic and Deterministic Dependencies

Hoifung Poon
University of Washington

(Joint work with Pedro Domingos)
Reasoning w. Probabilistic and Deterministic Dependencies

- Crucial for many real-world problems
- Especially for **statistical relational learning**
- **A long-standing challenge**
- **SAT solvers**: Not applicable w. probabilistic dependencies
- **Probabilistic inference**
  - **Approximate**: Determinism $\Rightarrow$ MCMC, Belief Propagation fail
  - **Exact**:
    - Recent work to leverage deterministic dependencies
    - Unlikely to scale to large real-world problems
    - Not applicable with near-deterministic dependencies
The MC-SAT Algorithm

- We developed **MC-SAT** to meet this challenge
- **MC-SAT = MCMC + SAT**
  - **MCMC**: Slice sampling w. an auxiliary variable for each clause
  - **SAT**: Wraps around SampleSAT (a uniform sampler) to sample from highly non-uniform distributions
- **Sound**: Satisfies ergodicity & detailed balance
- **Efficient**: Orders of magnitude faster than Gibbs and simulated tempering
- Accepts problems defined in **Markov logic**
Outline

- **Background**
  - Satisfiability (SAT)
  - Probabilistic inference
  - Markov logic
- The MC-SAT algorithm
- Experiments
- Conclusion
Satisfiability

- Find a truth assignment that satisfies all clauses
- The prototypical NP-complete problem
- Tremendous recent progress in efficient solvers
- E.g., WalkSAT [Selman et. al. 1996]
Probabilistic Inference

- Graphical model

\[
P(x) = \frac{1}{Z} \prod_{i} \Phi_i(x_{\{i\}})
\]

- Equivalently, log-linear model

\[
P(x) = \frac{1}{Z} \exp \left( \sum_{i} w_i f_i(x) \right)
\]

(Deterministic dependencies: \( w_i \to \infty \))

- **Problem**: Compute conditional probabilities
- **Widely used method**: Markov chain Monte Carlo (MCMC)
Markov Chain Monte Carlo

- **Gibbs sampling**
  - For each variable in turn, sample next value given its neighbors
  - Larger weights $\Rightarrow$ Exponentially slower convergence
  - Infinite weights (determinism): **Ergodicity is broken**

- **Simulated tempering**
  - Run parallel chains w. reduced weights, periodically swap chains
  - Large weights $\Rightarrow$ swaps very rare
  - Infinite weights: **Ergodicity still broken**
Auxiliary-Variable Methods

- Main ideas:
  - Use auxiliary variables to capture dependencies
  - Turn difficult sampling into uniform sampling
- Given distribution $P(x)$

$$f(x,u) = \begin{cases} 1, & \text{if } 0 \leq u \leq P(x) \\ 0, & \text{otherwise} \end{cases} \implies \int f(x,u) \, du = P(x)$$

- Sample from $f(x,u)$, then discard $u$
Slice Sampling [Damien et al. 1999]

\[ u^{(k)} \]

\[ u^{(k)} \]

\[ P(x) \]

\[ X^{(k)} \]

\[ X^{(k+1)} \]
Slice Sampling

- Identifying the slice may be difficult

\[ P(x) = \frac{1}{Z} \prod_i \Phi_i(x) \]

- Introduce an auxiliary variable \( u_i \) for each \( \Phi_i \)

\[ f(x, u_1, \cdots, u_n) = \begin{cases} 1 & \text{if } 0 \leq u_i \leq \Phi_i(x) \\ 0 & \text{otherwise} \end{cases} \]
Markov Logic

- A general language capturing logic and uncertainty
- A Markov Logic Network (MLN) is a set of pairs (F, w) where
  - F is a formula in first-order logic
  - w is a real number
- Together with constants, it defines a Markov network with
  - One node for each ground predicate
  - One feature for each ground formula F, with the corresponding weight w

\[
P(x) = \frac{1}{Z} \exp \left( \sum_i w_i f_i(x) \right)
\]
Outline

- Background
  - Satisfiability (SAT)
  - Probabilistic inference
  - Markov logic
- **The MC-SAT algorithm**
- Experiments
- Conclusion
The MC-SAT Algorithm

- Approximate inference for Markov logic
- Use slice sampling in MCMC
  - Auxiliary var. $u_i$ for each clause $C_i$: $0 \leq u_i \leq \exp(w_if_i(x))$
  - $C_i$ unsatisfied: $0 \leq u_i \leq 1$
    $\Rightarrow \exp(w_if_i(x)) \geq u_i$ for any next state $x$
  - $C_i$ satisfied: $0 \leq u_i \leq \exp(w_i)$
    $\Rightarrow$ With prob. $1 - \exp(-w_i)$, next state $x$ must satisfy $C_i$
    to ensure that $\exp(w_if_i(x)) \geq u_i$
The MC-SAT Algorithm

- Select random subset $M$ of satisfied clauses
- Larger $w_i \Rightarrow C_i$ more likely to be selected
- Hard clause ($w_i \rightarrow \infty$): Always selected
- Slice = States that satisfy clauses in $M$
- Sample uniformly from these
The MC-SAT Algorithm

\( X(0) \leftarrow \) A random solution satisfying all hard clauses

\textbf{for} \( k \leftarrow 1 \) \textbf{to} \( \text{num\_samples} \)

\[ M \leftarrow \emptyset \]

\textbf{forall} \( C_i \) satisfied by \( X(k - 1) \)

\hspace{1cm} \text{With prob.} \ 1 - \exp(-w_i) \text{ add } C_i \text{ to } M

\textbf{endfor}

\( X(k) \leftarrow \) A uniformly random solution satisfying \( M \)

\textbf{endfor}
The MC-SAT Algorithm

- **Sound**: Satisfies ergodicity and detailed balance (Assuming we have a perfect uniform sampler)

- Approximately uniform sampler [Wei et al. 2004]
  - **SampleSAT = WalkSAT + Simulated annealing**
  - **WalkSAT**: Find a solution very efficiently
    - But may be highly non-uniform
  - **Sim. Anneal.**: Uniform sampling as temperature $\rightarrow 0$
    - But very slow to reach a solution

- Trade off uniformity vs. efficiency
  by tuning the prob. of WS steps vs. SA steps
Experiments

- Domains
  - Entity resolution
  - Collective classification
- Methodology
- Results
Entity Resolution

Problem
- Which observations correspond to the same entity?
- E.g. Identify duplicate records when merging DBs
- Crucial for large scientific projects, businesses, etc.

Dataset
- BibServ: http://bibserv.org
- We used the CiteSeer and user-donated subset (88,035 citations)
- Generate dataset w. the largest canopies [McCallum et al. 2000]

MLN
- Evidence: Author / Title / Venue / TF-IDF scores for pairs
- Query: SameAuthor, SameVenue, SameTitle, SameBib
- Size:
  - 100 objs ⇒ 14 K query atoms, 1.25 million ground clauses
  - 150 objs ⇒ 30 K query atoms, 4 million ground clauses
Collective Classification

- **Problem**
  - Simultaneously classify a set of related objects
  - Many important instances:
    - Web page classification, image segmentation, spin glasses,
    - Social network analysis, word-of-mouth marketing, HMMs, etc.

- **Dataset**
  - Randomly generated w. varying number of objects and categories

- **MLN**
  - **Evidence**: Object attributes, attribute-category maps
  - **Query**: $C(x,u)$ ($x$ is of category $u$)
  - **Size**:
    - 100 objs $\Rightarrow$ 2 K query atoms, 105 K ground clauses
    - 150 objs $\Rightarrow$ 4.5 K query atoms, 349 K ground clauses
Methodology

- **System**: Extension to Alchemy [Kok et al. 2005]

- **Evaluation**: Log-loss of test data
  (Proxy for K-L divergence)
Results: Time

Entity Resolution

Collective Classification

Hard-clause weight = 1000
Results: Weight

Entity Resolution

Collective Classification

Maximum inference time: 100 minutes
Results: Number of Objects

Entity Resolution

Collective Classification

Maximum inference time: 100 minutes
Conclusion

- Reasoning w. probabilistic and deterministic dependencies is a long-standing challenge for AI.
- Particularly important in *statistical relational learning*.
- We developed **MC-SAT** to meet this challenge.
  
  **MC-SAT** = **MCMC (Slice Sampling)** + **SAT (SampleSAT)**

- **Sound** (Assuming perfect uniform sampler)
- **Efficient**
  - Experiments in entity resolution and collective classification
  - Orders of magnitude faster than Gibbs and tempering

- **Future work**: Use in learning; other apps; reduce memory