Statistical Relational Learning

- Relational tasks are everywhere
  - Collective classification (Tasker et al. 2002)
  - Information extraction (Poon & Domingos 2007; Bunescu 2004)
  - Social network modeling (Kemp et al. 2006)

- Modeling Long Range Dependency (LRD) is hard
  - e.g.字第entiful\&friends(B,C) ⇒smokes(C)
  - e.g. IsMotherOf(A,B) ⇒IsFatherOf(A,C)

- The Markov blanket of a variable grows prohibitively fast as the model’s order of Markov dependency grows.

- Discovering hidden roles help capture LRD
  - e.g. topic models, block models

- Thus reduce the need of extensive structure learning

Bayesian Networks vs. MRFs

- Relational Bayesian Networks
  - Easy to do inference and learning
  - Cannot learn structure automatically (acyclic constraint)

- Relational Markov Networks (RMNs)
  - Flexibility in representing complex patterns
  - Inference and learning are harder

Tree RMN

The Markov blanket of an entity e can be concisely defined by a relation tree starting from its type node

Variable Induction

- Adding variables as needed

Algorithm 1: Cumulative Variable Induction

- Initialize a new RMN $\mathcal{M} = (G, \theta)$
- While true do
  - Initialize parameters $\theta$ by L-UTGC
  - $\mathcal{M}$ = induced Hidden Variables (HV)
  - If no hidden variable is induced then
    - End
  - Else while return $\mathcal{M}$

How to efficiently evaluate a candidate $H$?

2nd order Taylor expansion estimates that each new feature $f$ brings maximum gain $\Delta_f = \frac{1}{2} \sum_{i \neq j} \epsilon_i (\theta_1) \delta_j (\theta_1) + \beta \delta_i (\theta_1)$

where $i$ is the set of entities with $H=1$, the overall gain is

$\Delta = \sum \Delta_f$

How to efficiently sift through all candidates?

We use a naive bottom up clustering algorithm

Algorithm 2: Bottom Up Clustering of Entities

- Initialize clustering $\mathcal{G} = (\mathcal{E} \cup \{\mathcal{C}\})$
- While true do
  - for each pair of clusters $I, J \subseteq \mathcal{E}$ do
    - $\text{inc}(I, J) = \Delta_{I+J} - \Delta_I - \Delta_J$
  - End for
  - if the largest increment $\leq 0$ then
    - End if
  - Merge the pair with the largest increment
  - End while

Results

- The datasets and a learning curve

Table 1: Number of entities (E) and attributes (A) for four datasets.

- $\mathcal{K}$ has only one attribute which has 76 possible values.

Example hidden variables

Animal data

- Entities
  - C0 KenSafetyWhite Sad Dolphin BlueWhite
  - Walnut HancockWhite
  - C1 GrizzlyBear Tiger GrizzlyShephard
  - Lappet Wolf Weasel Roosevelt Fox

- Positive Features
  - Flowers FowerSwim
  - Fish HaveCone genetic
  - Claws HaveClaw

- Negative Features
  - Grase Toekoh Hoosen
  - Strong Muscle Big Tooth

UML data

- Entities
  - A1 AccusedCriminal
  - Abnormalities
  - A2 Algo Plant

- Positive Features
  - E.g. Color: Green
  - E.g. Companions: $x$ is CompanionsWith $y$
  - E.g. LocationX: $x$ is LocatedX

- Negative Features
  - E.g. Property: $x$ is PropertyOf

Main result

- UML: $\lambda=0.05, 1$, $\delta=10$
  - Animal $\lambda=0.01, 1$, $\delta=10$

Previous approaches

- MLN structure learning (MLS) [10]
- Infinite Relational Models (IRM) [9]
- Multiple Relational Clustering (MRC) [11]