Outline

- Motivate problem
- Define PRMs
- Extensions and future work
Our Goal

- Observation: the world consists of many distinct entities with similar behaviors
- Exploit this redundancy to make our models simpler
- This was the idea of FOL: use quantification to eliminate redundant sentences over ground literals
Example: A simple domain

- a set of students, $\mathcal{S} = \{s_1, s_2, s_3\}$
- a set of professors, $\mathcal{P} = \{p_1, p_2, p_3\}$
- Well-Funded, Famous : $\mathcal{P} \rightarrow \{true, false\}$
- Student-Of : $\mathcal{S} \times \mathcal{P} \rightarrow \{true, false\}$
- Successful : $\mathcal{S} \rightarrow \{true, false\}$
Example: A simple domain

We can express a certain self-evident fact in one sentence of FOL:

$$\forall s \in S \quad \forall p \in P$$

Famous$(p)$ and Student-Of$(s, p)$

$$\Rightarrow \text{Successful}(s)$$
Example: A simple domain

The same sentence converted to propositional logic:

\[ \neg (p_{1\text{famous}} \text{ and student\_of\_s1\_p}_1 \text{ or } s_{1\text{successful}}) \text{ and } \neg (p_{1\text{famous}} \text{ and student\_of\_s2\_p}_1 \text{ or } s_{2\text{successful}}) \text{ and } \neg (p_{1\text{famous}} \text{ and student\_of\_s3\_p}_1 \text{ or } s_{3\text{successful}}) \text{ and } \neg (p_{2\text{famous}} \text{ and student\_of\_s1\_p}_1 \text{ or } s_{1\text{successful}}) \text{ and } \neg (p_{2\text{famous}} \text{ and student\_of\_s2\_p}_1 \text{ or } s_{2\text{successful}}) \text{ and } \neg (p_{2\text{famous}} \text{ and student\_of\_s3\_p}_1 \text{ or } s_{3\text{successful}}) \text{ and } \neg (p_{3\text{famous}} \text{ and student\_of\_s1\_p}_1 \text{ or } s_{1\text{successful}}) \text{ and } \neg (p_{3\text{famous}} \text{ and student\_of\_s2\_p}_1 \text{ or } s_{2\text{successful}}) \text{ and } \neg (p_{3\text{famous}} \text{ and student\_of\_s3\_p}_1 \text{ or } s_{3\text{successful}}) \]
Our Goal

- Unfortunately, the real world is not so clear-cut
- Need a probabilistic version of FOL
- Proposal: PRMs

Diagram:

Propositional Logic → Bayes Nets

Propositional Logic → PRMs

First-order Logic → PRMs
Defining the Schema

■ The world consists of base entities, partitioned into classes $X_1, X_2, ..., X_n$

■ Elements of these classes share connections via a collection of relations $R_1, R_2, ..., R_m$

■ Each entity type is characterized by a set of attributes, $A(X_i)$. Each attribute $A_j \in A(X_i)$ assumes values from a fixed domain, $V(A_j)$

■ Defines the *schema* of a relational model
Continuing the example...

We can modify the domain previously given to this new framework:

- 2 classes: $S, \mathcal{P}$
- 1 relation: $\text{Student-Of} \subseteq S \times \mathcal{P}$
- $\mathcal{A}(S) = \{\text{Success}\}$
- $\mathcal{A}(\mathcal{P}) = \{\text{Well-Funded, Famous}\}$
An instantiation $\mathcal{I}$ of the relational schema defines:

- a concrete set of base entities $\mathcal{O}^\mathcal{I}(X_i)$ for each class $X_i$
- values for the attributes of each base entity for each class
- $R_i(X_1, ..., X_k) \subset \mathcal{O}^\mathcal{I}(X_1) \times ... \times \mathcal{O}^\mathcal{I}(X_k)$ for each $R_i$. 
Slot chains

We can project any relation $R(X_1, ..., X_k)$ onto its $i$th and $j$th components to obtain a binary relation $\rho(X_i, X_j)$.

Consider this a function mapping instances of $X_i$ to sets of instances of $X_j$. That is, for $x \in O^T(X_i)$, let $x.\rho = \{y \in O^T(X_j) | (x, y) \in \rho(X_i, X_j)\}$.

We call $\rho$ a slot of $X_i$. Composition of slots (via transitive closure) gives a slot chain.
The idea of a PRM is to express a joint probability distribution over all possible instantiations of a particular relational schema.

Since there are infinitely many possible instantiations to a given schema, specifying the full joint distribution would be very painful.

Instead, compute marginal probabilities over remaining variables given a partial instantiation.
Partial Instantiations

A partial instantiation $I'$ specifies

- the sets $O^{I'}(X_i)$
- the relations $R_j$
- values of some attributes for some of the base entities

The PRM inference problem is to calculate the distribution over values for the unassigned attributes.
Locality of Influence

- BNs and PRMs are alike in that they both assume that real-world data exhibits *locality of influence*, the idea that most variables are influenced by only a few others.

- Both models exploit this property through *conditional independence*.

- PRMs go beyond BNs by assuming that there are few distinct patterns of influence in total
For a class $X$, values of the attribute $X.A$ are influenced by attributes in the set $Pa(X.A)$ (its parents).

$Pa(X.A)$ contains attributes of the form $X.B$ ($B$ an attribute) or $X.\tau.B$ ($\tau$ a slot chain).

As in a BN, the value of $X.A$ is conditionally independent of the values of all other attributes, given its parents.
An example

Captures the FOL sentence from before in a probabilistic framework.
Compiling into a BN

A PRM can be compiled into a BN, just as a statement in FOL can be compiled to a statement in PL.
We can use this network to support inference over queries regarding base entities.
Aggregates

- \( Pa(X.A) \) may contain \( X.\tau.B \) for slot chain \( \tau \), which is generally a multiset.
- \( Pa(X.A) \) dependent on the value of the set, not just the values in the multiset.
- Representational challenge, again \( |X.\tau.B| \) has no bound a priori.
Aggregates

- \( \gamma \) summarizes the contents of \( X.\tau.B \)
- Let \( \gamma(X.\tau.B) \) be a parent of attributes of \( X \)
- Many useful aggregates: mean, cardinality, median, etc.
- Require computation of \( \gamma \) to be deterministic (we can omit it from the diagram)
Example: Aggregates

- Let $\gamma(A) = |A|$
- Let $\text{Advisor-Of} = \text{Student-Of}^{-1}$
- e.g. $p_1.\text{Advisor-Of} = \{s_1\}$,
  $p_2.\text{Advisor-Of} = \{\}$,
  $p_3.\text{Advisor-Of} = \{s_2, s_3\}$

- To represent the idea that a professor’s funding is influenced by the number of advisees:

$$Pa(\mathcal{P}.\text{Well-Funded}) = \\
\{\mathcal{P}.\text{Famous}, \gamma(\mathcal{P}.\text{Advisor-Of})\}$$
Extensions

- Reference uncertainty. Not all relations known a priori; may depend probabilistically on values of attributes. E.g., students prefer advisors with more funding.

- Identity uncertainty. Distinct entities might not refer to distinct real-world objects.

- Dynamic PRMs. Objects and relations change over time; can be unfolded into a DBN at the expense of a very large state space.
Acknowledgements

“Approximate inference for first-order probabilistic languages” gives a promising MCMC approach for addressing relational and identity uncertainty.

“Inference in Dynamic Probabilistic Relational Models”, Sanhai et al. Particle-filter based DPRM inference that uses abstraction smoothing to generalize over related objects.