Query Evaluation on Probabilistic Databases

CSE 544: Wednesday, May 24, 2006
### Problem Setting

**Tables:**

<table>
<thead>
<tr>
<th>Movie</th>
<th>Review</th>
<th>Rating</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>title</td>
<td>name</td>
<td>rating</td>
<td>p</td>
</tr>
<tr>
<td>Twelve Monkeys</td>
<td>Monkey Love</td>
<td>good</td>
<td>.5</td>
</tr>
<tr>
<td>Monkey Love</td>
<td></td>
<td>fair</td>
<td>.2</td>
</tr>
<tr>
<td>Monkey Love</td>
<td></td>
<td>fair</td>
<td>.6</td>
</tr>
<tr>
<td>Monkey Love Pl</td>
<td></td>
<td>poor</td>
<td>.9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>title</th>
<th>year</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>Twelve Monkeys</td>
<td>1995</td>
<td>.8</td>
</tr>
<tr>
<td>Monkey Love</td>
<td>1997</td>
<td>.4</td>
</tr>
<tr>
<td>Monkey Love</td>
<td>1935</td>
<td>.9</td>
</tr>
<tr>
<td>Monkey Love Pl</td>
<td>2005</td>
<td>.7</td>
</tr>
</tbody>
</table>

**Queries:**

\[
A(x,y) \leftarrow \text{Review}(x,y), \quad \text{Movie}(x,z), \ z > 1991
\]

**Answers:**

<table>
<thead>
<tr>
<th>title</th>
<th>rating</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>Twelve Monkeys</td>
<td>fair</td>
<td>.53</td>
</tr>
<tr>
<td>Monkey Love</td>
<td>good</td>
<td>.42</td>
</tr>
<tr>
<td>Monkey Love Pl</td>
<td>fair</td>
<td>.15</td>
</tr>
</tbody>
</table>
Two Problems

Fixed schema $S$, conjunctive query $Q(x,y)$

Query evaluation problem

Fix answer tuple $(a,b)$
Given database $I$, compute $\Pr(Q(a,b))$

Top-$k$ answering problem

Fix $k > 0$
Given database $I$, find $k$ answer tuples with highest probabilities
Related Work: DB

- Cavallo & Pitarelli: 1987
- Barbara, Garcia-Molina, Porter: 1992
- Lakshmanan, Leone, Ross & Subrahmanian: 1997
- Fuhr & Roellke: 1997
- Dalvi & S: 2004
- Widom: 2005
Related Work: Logic

- Degrees of belief [Bacchus, Grove, Halpern, Koller’96]
- Probabilistic Logic [Nielson]
- Probabilistic model checking [Kwiatkowska’02]
- Probabilistic Relational Model [Taskar, Abbeel, Koller’02]
Probabilistic Database

Schema S, Domain D, Set of instances Inst

Definition
Probabilistic database is a probability distribution

\[ Pr : \text{Inst} \rightarrow [0,1], \quad \sum_I Pr[I] = 1 \]

If \( Pr[I] > 0 \) then I is called “possible world”
Probabilistic Database

Representation:

- **Independent tuples:**
  - I-database DB over some schema $S^i$

- **Independent and disjoint tuples:**
  - ID-database DB over some schema $S^{id}$

Semantics:

- DB “means” probability distribution $Pr$ over schema $S$
Independent Events

- A tuple is in the database with probability $p$
- Any two tuples are independent events
Representation

I-Databases

Reviews \( i(M,S,p) \)

<table>
<thead>
<tr>
<th>Movie</th>
<th>Score</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>m42</td>
<td>good</td>
<td>( P_1 )</td>
</tr>
<tr>
<td>m99</td>
<td>good</td>
<td>( P_2 )</td>
</tr>
<tr>
<td>m76</td>
<td>poor</td>
<td>( P_3 )</td>
</tr>
</tbody>
</table>

Reviews \( (M,S) \)

\[
Pr[I_1] = (1-p_1)(1-p_2)(1-p_3)
\]

\[
Pr[I_4] = P_1P_2(1-p_3)
\]

\[
Pr[I_1] + Pr[I_2] + \ldots + Pr[I_8] = 1
\]

Possible worlds semantics
Disjoint Events

Needed in

- Many-to-1 matchings
- Possible values for attributes [Barbara’92]

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>34</td>
<td>0.3</td>
</tr>
<tr>
<td></td>
<td>43</td>
<td>0.7</td>
</tr>
<tr>
<td>Mary</td>
<td>25</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>34</td>
<td>0.3</td>
</tr>
<tr>
<td></td>
<td>43</td>
<td>0.7</td>
</tr>
<tr>
<td>Mary</td>
<td>25</td>
<td>1.0</td>
</tr>
</tbody>
</table>
### ID-Databases

<table>
<thead>
<tr>
<th>Time&lt;sup&gt;d&lt;/sup&gt;</th>
<th>Activity</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>t</td>
<td>walk</td>
<td>P&lt;sub&gt;1&lt;/sub&gt;</td>
</tr>
<tr>
<td>t</td>
<td>run</td>
<td>P&lt;sub&gt;2&lt;/sub&gt;</td>
</tr>
<tr>
<td>t+1</td>
<td>walk</td>
<td>P&lt;sub&gt;3&lt;/sub&gt;</td>
</tr>
</tbody>
</table>

**Activities<sup>id</sup>**

\[
\sum_{i=1}^{6} \Pr[I_i] = 1
\]

\[
\Pr[I_1] = (1-p_1-p_2)(1-p_3)
\]

\[
\Pr[I_3] = p_2(1-p_3)
\]

\[
\Pr[I_5] = p_1p_3
\]

\[
\Pr[I_1] + \Pr[I_2] + \ldots + \Pr[I_6] = 1
\]
ID subsumes $I$

**Reviews$^i$**

<table>
<thead>
<tr>
<th>Movie</th>
<th>Score</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>m42</td>
<td>good</td>
<td>$P_1$</td>
</tr>
<tr>
<td>m99</td>
<td>good</td>
<td>$P_2$</td>
</tr>
<tr>
<td>m76</td>
<td>poor</td>
<td>$P_3$</td>
</tr>
</tbody>
</table>

**Reviews$^id$**

<table>
<thead>
<tr>
<th>Movie$^d$</th>
<th>Score$^d$</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>m42</td>
<td>good</td>
<td>$P_1$</td>
</tr>
<tr>
<td>m99</td>
<td>good</td>
<td>$P_2$</td>
</tr>
<tr>
<td>m76</td>
<td>poor</td>
<td>$P_3$</td>
</tr>
</tbody>
</table>

**Note:**

**Reviews$^id$**

<table>
<thead>
<tr>
<th>Movie$^d$</th>
<th>Score$^d$</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>m42</td>
<td>good</td>
<td>$P_1$</td>
</tr>
<tr>
<td>m99</td>
<td>good</td>
<td>$P_2$</td>
</tr>
<tr>
<td>m76</td>
<td>poor</td>
<td>$P_3$</td>
</tr>
</tbody>
</table>

means all tuples are disjoint
Queries
Syntax: conjunctive queries over schema S

\[ Q(y) :- \text{Movie}(x,y), \text{Review}(x,z), z \geq 3 \]
Two Query Semantics

Possible answer sets

- Given set A: \( \Pr\{\{t \mid I \models Q(t)\} = A\} \)
- Used for views

Possible tuples

- Given tuple t: \( \Pr[I \models Q(t)] \)
- Used for query evaluation and top-k
### Query Semantics

**Tuple probabilities**

<table>
<thead>
<tr>
<th>year</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>1935</td>
<td>$p_2 + p_3 = 0.6$</td>
</tr>
<tr>
<td>2004</td>
<td>$p_1 + p_3 = 0.5$</td>
</tr>
<tr>
<td>1995</td>
<td>$p_3 = 0.2$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Q(y) :- Movie(x,y), Review(x,z)**

<table>
<thead>
<tr>
<th>id</th>
<th>year</th>
</tr>
</thead>
<tbody>
<tr>
<td>m42</td>
<td>2004</td>
</tr>
<tr>
<td>m99</td>
<td>1901</td>
</tr>
<tr>
<td>m76</td>
<td>1902</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>id</th>
<th>year</th>
</tr>
</thead>
<tbody>
<tr>
<td>m99</td>
<td>1935</td>
</tr>
<tr>
<td>m05</td>
<td>1903</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>id</th>
<th>year</th>
</tr>
</thead>
<tbody>
<tr>
<td>m76</td>
<td>1995</td>
</tr>
<tr>
<td>m05</td>
<td>2004</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>id</th>
<th>year</th>
</tr>
</thead>
<tbody>
<tr>
<td>m87</td>
<td>1934</td>
</tr>
<tr>
<td>m44</td>
<td>1904</td>
</tr>
</tbody>
</table>

**top k**
Summary on Data Model

- **Data Model:**
  - Semantics = possible worlds
  - Syntax = I-databases or ID-databases

- **Queries:**
  - Syntax = unchanged (conjunctive queries)
  - Semantics = tuple probabilities
Problem Definition

Fix schema $S$, query $Q$, answer tuple $t$

Problem: given I/ID-database $DB$, compute $\Pr[I \models Q(t)]$

notation: $\Pr[Q(t)]$

Conventions:
For upper bounds ($P$ or $\#P$): probabilities are rationals
For lower bounds ($\#P$): probabilities are $1/2$
Query Evaluation on I-Databases

Outline

- Intuition
- Extensional plans: PTIME case
- Hard queries: #P-complete case
- Dichotomy Theorem
Q(y) :- Movie(x,y), Review(x,z)

**Intuition**

**Movie**

<table>
<thead>
<tr>
<th>id</th>
<th>year</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>m42</td>
<td>1995</td>
<td>$p_1$</td>
</tr>
<tr>
<td>m99</td>
<td>2002</td>
<td>$p_2$</td>
</tr>
<tr>
<td>m76</td>
<td>2002</td>
<td>$p_3$</td>
</tr>
<tr>
<td>m05</td>
<td>2005</td>
<td>$p_4$</td>
</tr>
</tbody>
</table>

**Review**

<table>
<thead>
<tr>
<th>mid</th>
<th>rate</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>m42</td>
<td>4</td>
<td>$q_1$</td>
</tr>
<tr>
<td>m42</td>
<td>2</td>
<td>$q_2$</td>
</tr>
<tr>
<td>m42</td>
<td>3</td>
<td>$q_3$</td>
</tr>
<tr>
<td>m99</td>
<td>1</td>
<td>$q_4$</td>
</tr>
<tr>
<td>m99</td>
<td>3</td>
<td>$q_5$</td>
</tr>
<tr>
<td>m76</td>
<td>5</td>
<td>$q_6$</td>
</tr>
</tbody>
</table>

**Answer**

<table>
<thead>
<tr>
<th>Year</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>1995</td>
<td>$p_1 \times (1 - (1 - q_1) \times (1 - q_2) \times (1 - q_3))$</td>
</tr>
<tr>
<td>2002</td>
<td>$1 - (1 - p_2 \times (1 - (1 - q_4) \times (1 - q_5)) \times (1 - p_3 \times q_6))$</td>
</tr>
</tbody>
</table>
I-Extensional Plans

[Barbara92,Lakshmanan97]

- Add \( p \)
- Join \( \bowtie \) \( p = p_1 \times p_2 \)
- Projection \( \Pi \) \( p = 1 - (1-p_1)(1-p_2)\ldots(1-p_n) \)
- Selection \( \sigma \) \( p = p \)

Note: data complexity is PTIME
Extensional Query Plans

\[ x \times x' \times p \times q \]

\[ \times \quad \times \quad 1-(1-p1)(1-p2)(1-p3) \]

\[ x \quad \times \quad p \]

\[ \sigma \]

\[ \times \quad \times \quad \times \quad \times \quad \times \quad p \]

\[ \times \quad \times \quad \times \quad d1 \]

\[ \times \quad \times \quad \times \quad d2 \]

\[ \times \quad \times \quad \times \quad d3 \]

\[ \times \quad \times \quad \times \quad p \]
Extensional Query Plans

- Each tuple \( t \) has a probability \( t.P \)
- Algebra operators compute \( t.P \)
- Data complexity: PTIME
\[ Q(y) :\text{-} \text{Movie}(x,y), \text{Review}(x,z) \]

\[ 1995 \cdot 1-(1-pq_1)(1-pq_2)(1-pq_3) \]

\[ 1995 \cdot p(1-(1-q_1)(1-q_2)(1-q_3)) \]

\[ m1 \cdot 1 - (1-q_1)(1-q_2)(1-q_3) \]
Theorem: Data complexity is \#P-complete

#P-Complete Queries

\[ Q_{bad} :- R^i(x), S(x,y), T^i(y) \]
Proof:

Theorem [Provan&Ball83] Counting the number of satisfying assignments for bipartite DNF is \#P-complete

Reduction:

\[ x_2y_3 \lor x_1y_2 \lor x_4y_3 \lor x_3y_1 \]

\[ Q_{bad} :\leftarrow R^i(x), S(x,y), T^i(y) \]
I-Dichotomy

Q = boolean conjunctive query

Definition 1. For each variable x:
\[ \text{goals}(x) = \text{set of goals that contain } x \]

Definition 2. Q is hierarchical if for all x, y:
(a) \[ \text{goals}(x) \cap \text{goals}(y) = \emptyset \], or
(b) \[ \text{goals}(x) \subseteq \text{goals}(y) \], or
(c) \[ \text{goals}(y) \subseteq \text{goals}(x) \]
Q :- R(x), S(x,y), T(x,y,z), K(x,v)

“hierarchical”

Q :- R(x), S(x,y), T(y)

“non-hierarchical”
I-Dichotomy

Schema $S^i = \{R^i_1, R^i_2, \ldots, R^i_m\}$

**Theorem** Let $Q = \text{conjunctive query w/o self-joins}$. Then one of the following holds:

- $Q$ is in PTIME
- $Q$ has a correct extensional plan
- $Q$ is hierarchical

or:

- $Q$ is \#P-complete
- $Q$ has subgoals $R(x,\ldots), S(x,y,\ldots), T(y,\ldots)$
Lemma 1.
If Q is non-hierarchical, then \#P-complete

Proof:

Q ::= \text{rest is like for } Q_{bad}
Lemma 2. If $Q$ is hierarchical, then $\text{PTIME}$

Proof:

Case 1: has no root

$\Pr(Q) = \Pr(Q_1) \Pr(Q_2) \Pr(Q_3)$

This is extensional join $\bowtie$
Proof

Case 2: has root $x$

$\text{Dom} = \{a_1, a_2, \ldots, a_n\}$

$\Pr(Q) = 1 - (1-\Pr(Q(a_1/x))(1-\Pr(Q(a_2/x))\ldots(1-\Pr(Q(a_n/x))))$

This is an extensional projection: $\Pi$

QED
Query Evaluation on ID-Databases

- ID-extensional plans
- \#P-complete queries
- Dichotomoy Theorem
Only difference: two kinds of projections:

independent \( 1-(1-p_1)\cdots(1-p_n) \)

disjoint \( p_1 + \cdots + p_n \)
#P-Complete Queries

Q₁ :- Rᵢ(x), Sᵢ(x,y), Tᵢ(y)

Q₂ :- Rᵈ(xᵈ,y), Sᵈ(yᵈ)

Q₃ :- Rᵈ(xᵈ,y), Sᵈ(zᵈ,y)
Theorem Let $Q = \text{conjunctive query w/o self-joins}$. Then one of the following holds:

- $Q$ is in PTIME
- $Q$ has a correct extensional plan

or:

- $Q$ is #P-complete
- $Q$ has one of $Q_1$, $Q_2$, $Q_3$ as subqueries
Extensions

Extensions of the dichotomy theorem exists for:

- Mixed schemas (some relations are deterministic)
- Functional dependencies
Summary on Query Evaluation

Extensional plans: popular, efficient, BUT

- “Equivalent” plans lead to different results
- Some queries admit “correct” plans

Some simple queries: #P-complete complexity

Dichotomy theorem

Future work: remove ‘no-self-join’ restriction
Conclusions

- Strong motivation from practical applications
  - Merge query and search technologies

- Probabilistic DB's are hard!
  - Hacks don’t work (yet). Need principled approach.
Thank you!

Questions?