Logic and Databases

Phokion G. Kolaitis

UC Santa Cruz & IBM Research - Almaden

Lecture 4 - Part 2
Alternative Semantics of Queries

- **Bag Semantics**
  
  We focused on the containment problem for conjunctive queries under bag semantics.

  Next, we will discuss:

- **Probabilistic Databases**

- **Inconsistent Databases**

  The focus will be on the data complexity of conjunctive queries in these two frameworks.
Probabilistic Databases

- So far, data stored in a database have been assumed to exist with certainty.
- However, in modern applications, data may be uncertain: noisy, fuzzy, corrupted, or even missing.
  - Such applications include social media, information integration, scientific data management, ...
- Probabilistic Databases provide a framework for modeling and managing uncertain data.
  - Probabilistic Databases extend relational databases with probabilities.
  - Both the data and their probabilities are stored as "standard" relations, but the semantics of query answering takes probabilities into account.
Probabilistic Databases

Definition
A probabilistic database is a pair $W = (D, P)$ such that

- $D = \{D_1, \ldots, D_k\}$ is a finite set of databases $D_i$ over the same schema.
- $P : D \rightarrow [0, 1]$ is a function such that $\sum_{i=1}^{k} P(D_k) = 1$.

Intuition
- A probabilistic database can be in one of finitely many possible states, each with some probability.
- $D$ is a set of possible worlds representing the possible states of the probabilistic database.
Marginal Probabilities

Definition
Let $W = (D, P)$ be a probabilistic database.

- Let $q$ be a $k$-ary query, $k \geq 1$, and let $a$ be a $k$-tuple. The marginal probability $Pr(q, a, W)$ of $a$ is

$$Pr(q, a, W) = \sum_{a \in q(D_i)} P(D_i).$$

- Let $q$ be a Boolean query. The marginal probability $Pr(q, W)$ of $q$ is

$$Pr(q, W) = \sum_{D_i \models q} P(D_i).$$
Query Evaluation over probabilistic databases: Given a $k$-ary query $q$, a $k$-tuple $a$, and a probabilistic database $W$, compute the marginal probability $Pr(q, a, W)$.

Note that this is a combined complexity problem. Here, we will focus on the data complexity of Boolean conjunctive queries over probabilistic databases.

Fix a Boolean conjunctive query $q$. Then $Pr[q]$ is the following algorithmic problem: Given a probabilistic database $W$, compute the marginal probability $Pr(q, W)$. 
A probabilistic database may have an arbitrarily large number of possible worlds, which implies that listing all these possible worlds may be infeasible.

For this reason, several different compact representations of probabilistic databases have been introduced and investigated.

Here, we will focus on tuple-independent databases, which is arguably the simplest model for probabilistic database design.

Intuitively, in a tuple-independent database all tuples are independent probabilistic events.
Tuple-Independent Databases

- A tuple-independent relation is a relation \( R(A_1, \ldots, A_m, P) \) in which tuples \((a_1, \ldots, a_m)\) are independent events and the values of \( P \) are numbers in the interval \([0, 1]\) denoting the marginal tuple probabilities of the tuples.

<table>
<thead>
<tr>
<th>Company</th>
<th>Product</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple</td>
<td>iphone 6</td>
<td>0.95</td>
</tr>
<tr>
<td>Samsung</td>
<td>Galaxy 7</td>
<td>0.96</td>
</tr>
<tr>
<td>Apple</td>
<td>iphone 7</td>
<td>0.75</td>
</tr>
<tr>
<td>Microsoft</td>
<td>Lumia 640</td>
<td>0.85</td>
</tr>
</tbody>
</table>

- This table is a compact representation of 16 possible tables.
- For example, the table consisting of the first, the second, and the fourth tuple has probability \(0.95 \cdot 0.96 \cdot 0.25 \cdot 0.85\).

- A tuple-independent database is a database consisting of tuple-independent relations.
Fix a Boolean query \( q \).

\( Pr[q] \) is the following problem: Given a tuple-independent database \( W \), compute the marginal probability \( Pr(q, W) \).

This is a data complexity problem because the query is fixed and the input is a tuple-independent database \( W \).

Recall that the data complexity of unions of conjunctive queries on (deterministic) databases is in LOGSPACE.
Dichotomy Theorem (Dalvi and Suciu - 2012)

If $q$ is a union of Boolean conjunctive queries, then $Pr[q]$ is in $P$ or $Pr[q]$ is $\#P$-complete.
Dichotomy Theorem (Dalvi and Suciu - 2012)
If $q$ is a union of Boolean conjunctive queries, then $Pr[q]$ is in P or $Pr[q]$ is \#P-complete.

Note

- \#P is the class of counting problems associated with decision problems in NP.
- The prototypical \#P-complete problem is \#SAT: Given a CNF-formula $\varphi$, compute the number of its satisfying assignments.
- Valiant (1979) also showed that \#POSITIVE 2SAT is \#P-complete. ("easy" decision - "hard" counting phenomenon)
Hierarchical Queries

Definition

- A self-join free conjunctive query is a conjunctive query in which no relation symbol appears more than once.
- Let \( q \) be a self-join free conjunctive query.
  - If \( x \) is a variable of \( q \), then \( at(x) \) is the set of all atoms of \( x \) in which \( x \) appears.
  - We say that \( q \) is hierarchical if for every two variables \( x \) and \( y \) of \( q \), one of the following holds:
    \[
    at(x) \subseteq at(y), \quad at(y) \subseteq at(x), \quad at(x) \cap at(y) = \emptyset.
    \]

Example

- The query \( \exists x \exists y (R(x) \land S(x, y)) \) is hierarchical.
- The query \( \exists x \exists y (R(x) \land S(x, y) \land T(y)) \) is not hierarchical.
The Little Dichotomy Theorem (Dalvi and Suciu - 2004)

Let $q$ be a Boolean self-join free conjunctive query.

- If $q$ is hierarchical, then $Pr[q]$ is in $P$.
- If $q$ is not hierarchical, then $Pr[q]$ is $\#P$-complete.
The Little Dichotomy Theorem (Dalvi and Suciu - 2004)
Let $q$ be a Boolean self-join free conjunctive query.

- If $q$ is hierarchical, then $Pr[q]$ is in $P$.
- If $q$ is not hierarchical, then $Pr[q]$ is $\sharp P$-complete.

Proof Idea

- Hierarchical queries admit safe evaluation plans.
- Non-hierarchical queries:
  - Show that $Pr[\exists x \exists y (R(x) \land S(x, y) \land T(y))]$ is $\sharp P$-complete.
  - Show that if $q$ is not hierarchical, then $Pr[\exists x \exists y (R(x) \land S(x, y) \land T(y))]$ is reducible to $Pr[q]$. 
Hierarchical Queries

Let \( q \) be the hierarchical query \( \exists x \exists y (R(x) \land S(x, y)) \) and let \( W \) be a tuple-independent database.

- First, write \( q \) as \( \exists x (R(x) \land \exists y S(x, y))\).
- Then, using tuple-independence repeatedly, we have that:

\[
Pr[q] = 1 - \prod_{a \in \text{adom}(W)} \left( 1 - P((R(a) \land \exists y S(a, y))) \right)
\]

\[
= 1 - \prod_{a \in \text{adom}(W)} \left( 1 - P((R(a)) \cdot P(\exists y S(a, y))) \right)
\]

\[
= 1 - \prod_{a \in \text{adom}(W)} \left( 1 - P((R(a)) \cdot (1 - \prod_{b \in \text{adom}(W)} (1 - P(S(a, b)))))) \right)
\]

- The last expression has size \( O(n^2) \), where \( n = |\text{adom}(W)| \).
A positive partitioned 2DNF formula (PP2DNF) is a DNF-formula of the form

\[ x_{i_1} y_{j_1} \lor \cdots \lor x_{i_k} y_{j_k}, \]

where the \( x_i \)'s and the \( y_j \)'s form disjoint sets of variables.

Theorem (Provan and Ball - 1982)
\( \#\text{PP2DNF} \) is \#P-complete.

Theorem (Dalvi and Suciu - 2004)
There is a counting reduction from \( \#\text{PP2DNF} \) to \( \text{Pr}[\exists x \exists y (R(x) \land S(x, y) \land T(y))] \).
Non-Hierarchical Queries

Counting reduction from \#PP2DNF to \( \text{Pr}[\exists x \exists y (R(x) \land S(x, y) \land T(y))] \).

- Suppose \( \varphi \) is the formula \( x_1 y_1 \lor x_1 y_2 \lor x_2 y_1 \).
- Let \( W_{\varphi} \) be the tuple-independence database:

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( R )</td>
<td>( X )</td>
<td>( P )</td>
<td>( S )</td>
<td>( X )</td>
<td>( Y )</td>
<td>( P )</td>
<td>( T )</td>
</tr>
<tr>
<td>( x_1 )</td>
<td>0.5</td>
<td></td>
<td>( x_1 )</td>
<td>( y_1 )</td>
<td>1</td>
<td></td>
<td>( y_1 )</td>
</tr>
<tr>
<td>( x_2 )</td>
<td>0.5</td>
<td></td>
<td>( x_1 )</td>
<td>( y_2 )</td>
<td>1</td>
<td></td>
<td>( y_2 )</td>
</tr>
<tr>
<td>( x_2 )</td>
<td></td>
<td>( x_1 )</td>
<td>( y_1 )</td>
<td>( x_2 )</td>
<td>( y_1 )</td>
<td>( x_2 )</td>
<td>( y_1 )</td>
</tr>
</tbody>
</table>

- There is a 1-1 correspondence between truth assignments for \( \varphi \) and possible worlds for \( W_{\varphi} \).
- It is easy to see that \( \#\varphi = 2^n \text{Pr}(\exists x \exists y (R(x) \land S(x, y) \land T(y)), W_{\varphi}), \) where \( n \) is the number of variables of \( \varphi \).
Non-Hierarchical Queries

Let \( q \) be a Boolean conjunctive query that is not hierarchical.

- By definition, there are variables \( x \) and \( y \) of \( q \) such that \( \text{at}(x) \not\subseteq \text{at}(y), \text{at}(y) \not\subseteq \text{at}(x), \text{at}(x) \cap \text{at}(y) \neq \emptyset \).

- Since \( \text{at}(x) \not\subseteq \text{at}(y) \), there is an atom \( R'(x, \ldots) \) in which \( y \) does not appear.

- Since \( \text{at}(y) \not\subseteq \text{at}(x) \), there is an atom \( T'(y, \ldots) \) in which \( x \) does not appear.

- Since \( \text{at}(x) \cap \text{at}(y) \neq \emptyset \), there is an atom \( T'(x, y, \ldots) \) in which both \( x \) and \( y \) appear.

- These atoms can be used to obtain a counting reduction from \( Pr[\exists x \exists y (R(x) \land S(x, y) \land T(y))] \) to \( Pr[q] \).
Data Complexity in Tuple-Independent Databases

The Little Dichotomy Theorem (Dalvi and Suciu - 2004)
Let $q$ be a Boolean self-join free conjunctive query.

- If $q$ is hierarchical, then $Pr[q]$ is in $P$.
- If $q$ is not hierarchical, then $Pr[q]$ is $\#P$-complete.

Open Problems:
- Dichotomy Theorem for arbitrary conjunctive queries on the block-independent-disjoint model.
- Dichotomy known for self-join free conjunctive queries.
- Dichotomy Theorem for arbitrary conjunctive queries on the tuple-independent model in the presence of functional dependencies.
- Dichotomy known for self-join free conjunctive queries.
The Little Dichotomy Theorem (Dalvi and Suciu - 2004)
Let $q$ be a Boolean self-join free conjunctive query.

- If $q$ is hierarchical, then $Pr[q]$ is in $P$.
- If $q$ is not hierarchical, then $Pr[q]$ is $\#P$-complete.

Open Problems:

- Dichotomy Theorem for arbitrary conjunctive queries on the block-independent-disjoint model.
  - Dichotomy known for self-join free conjunctive queries.

- Dichotomy Theorem for arbitrary conjunctive queries on the tuple-independent model in the presence of functional dependencies.
  - Dichotomy known for self-join free conjunctive queries.