Modelling Retrieval Models in Probabilistic Logical Models

CSIRO, Canberra, 9th April 2008
Thomas Roelleke
Queen Mary University of London
Outline

- Introduction
- Background: ProbDB, IR-on-DB, Retrieval Models, PRA
- The Relational Bayes
  - DB+IR Toy Database:
    - person(Name, City, Nationality); coll(Term, DocId);
  - Motivation
  - City.Nationality and Nationality.City
  - Syntax and Examples
- Modelling Retrieval Models: TF-IDF, BIR, LM
- Implementation
- Summary and Outlook
Data and information independence

- Application: all-in-one
- User Interface
- Business Logic: "all-in-one"
- SQL
- Data

- Business Logic
- Ranking Information
- Data

- User Interface

Layers:
- all-in-one
- three-tier
- four-tier
Generation of probabilistic databases

Non-probabilistic database \[\xrightarrow{\text{Bayes}}\] Probabilistic database
## Layers

<table>
<thead>
<tr>
<th>Layer</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>External layer</td>
<td>Information sorted by relevance</td>
</tr>
<tr>
<td>Logical layer</td>
<td>Probabilistic relations</td>
</tr>
<tr>
<td></td>
<td>Probability estimation: Bayes<a href=""></a></td>
</tr>
<tr>
<td>Physical layer</td>
<td>Relational model/algebra; SQL</td>
</tr>
</tbody>
</table>
Probabilistic Databases

[Cavallo and Pittarelli, 1987]: Relational and probabilistic databases, information content (Shannon), probabilistic data dependencies

[Fuhr and Roelleke, 1997]: Probabilistic relational algebra for the integration of IR and DB: intensional vs extensional semantics, event expressions, solve Norbert’s "db AND NOT ir OR ir AND NOT db"

[Dalvi and Suciu, 2004]: Efficient query evaluation: intensional semantics and possible worlds semantics, safe-plan optimisation algorithm
IR-on-DB

[Schek and Pistor, 1982]: Data Structures for an Integrated Database Management and Information Retrieval System

[Agrawal et al., 2002, Hristidis and Papakonstantinou, 2002]: DBXplorer, DISCOVER: keyword search

[Schefe, 1983]: Natuerlichsprachiger Zugang zu Datenbanken?

[Chaudhuri et al., 2004, Chaudhuri et al., 2006]: Probabilistic ranking of database query results (based on BIR model)

[Ercegovac et al., 2005]: TEXTURE benchmark (automatic scaling of benchmark; three competing systems)

[Cornacchia and de Vries, 2007]: A parametrised search system
(Probabilistic) Retrieval Models

[Robertson and Sparck Jones, 1976, Croft and Harper, 1979]: BIR, “the probabilistic model”

[Wong and Yao, 1995, Roelleke et al., 2006]: $P(d \rightarrow q)$ and matrix framework


[Roelleke, 2003, Robertson, 2004, Robertson, 2005, Roelleke and Wang]: “probability of being informative”, on theoretical arguments for idf, on event spaces, a parallel derivation of probabilistic models


[Robert Ross, 2002]: probabilistic aggregates: the aggregates underline the difference between “normal” attributes and tuple probabilities

[Grossman and Frieder, 2004]: implement TF-IDF VSM in SQL

[Roelleke and Fuhr, 1996, Fuhr et al., 1998, Lalmas et al., 2002]: POOL SIGIR 06, “Dolores” SIGIR 98, POOL (in Intelligent Exploration of the Web 02), Logic in IR,
<table>
<thead>
<tr>
<th>Term</th>
<th>DocId</th>
</tr>
</thead>
<tbody>
<tr>
<td>sailing</td>
<td>doc1</td>
</tr>
<tr>
<td>boats</td>
<td>doc1</td>
</tr>
<tr>
<td>sailing</td>
<td>doc2</td>
</tr>
<tr>
<td>sailing</td>
<td>doc2</td>
</tr>
<tr>
<td>boats</td>
<td>doc2</td>
</tr>
<tr>
<td>sailing</td>
<td>doc3</td>
</tr>
<tr>
<td>east</td>
<td>doc3</td>
</tr>
<tr>
<td>coast</td>
<td>doc3</td>
</tr>
<tr>
<td>sailing</td>
<td>doc4</td>
</tr>
<tr>
<td>boats</td>
<td>doc5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>probColl: “tf”</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prob</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td>0.5</td>
</tr>
<tr>
<td>0.5</td>
</tr>
<tr>
<td>0.66</td>
</tr>
<tr>
<td>0.33</td>
</tr>
<tr>
<td>0.33</td>
</tr>
<tr>
<td>0.33</td>
</tr>
<tr>
<td>0.33</td>
</tr>
<tr>
<td>1.0</td>
</tr>
<tr>
<td>1.0</td>
</tr>
</tbody>
</table>
Thomas Roelleke

Modelling Retrieval Models
### Probabilistic Databases

**IR-on-DB**

(Probabilistic) Retrieval Models

PRA, PD, POOL

**Example**

<table>
<thead>
<tr>
<th>non-distinct</th>
<th>Prob</th>
<th>DocId</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.4 · 0.5</td>
<td></td>
<td>doc1</td>
</tr>
<tr>
<td>0.6 · 0.5</td>
<td></td>
<td>doc1</td>
</tr>
<tr>
<td>0.4 · 0.66</td>
<td></td>
<td>doc2</td>
</tr>
<tr>
<td>0.6 · 0.33</td>
<td></td>
<td>doc2</td>
</tr>
<tr>
<td>0.4 · 0.33</td>
<td></td>
<td>doc3</td>
</tr>
<tr>
<td>0.4 · 1.0</td>
<td></td>
<td>doc4</td>
</tr>
<tr>
<td>0.6 · 1.0</td>
<td></td>
<td>doc5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>distinct (aggregated)</th>
<th>Prob</th>
<th>DocId</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td></td>
<td>doc1</td>
</tr>
<tr>
<td>1.0</td>
<td></td>
<td>doc2</td>
</tr>
<tr>
<td>0.4 · 0.33</td>
<td></td>
<td>doc3</td>
</tr>
<tr>
<td>0.4</td>
<td></td>
<td>doc4</td>
</tr>
<tr>
<td>0.6</td>
<td></td>
<td>doc5</td>
</tr>
</tbody>
</table>

Aggregation of non-distinct tuples:

```sql
1 SELECT DISJOINT DocId FROM retrieved;
```
Where do the probabilities in probColl ("tf") and probQuery come from?

How can the estimation be described algebraically?
### Motivation

#### Toy Database

<table>
<thead>
<tr>
<th>person</th>
<th>coll</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>City</td>
</tr>
<tr>
<td>Peter</td>
<td>London</td>
</tr>
<tr>
<td>Paul</td>
<td>London</td>
</tr>
<tr>
<td>Mary</td>
<td>London</td>
</tr>
<tr>
<td>Thomas</td>
<td>Dortmund</td>
</tr>
<tr>
<td>Thomas</td>
<td>London</td>
</tr>
<tr>
<td>Thomas</td>
<td>Hamburg</td>
</tr>
<tr>
<td>Hany</td>
<td>London</td>
</tr>
<tr>
<td>Hany</td>
<td>London</td>
</tr>
<tr>
<td>Jun</td>
<td>London</td>
</tr>
<tr>
<td>Zhi</td>
<td>London</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Term</th>
<th>DocId</th>
</tr>
</thead>
<tbody>
<tr>
<td>sailing</td>
<td>doc1</td>
</tr>
<tr>
<td>boats</td>
<td>doc1</td>
</tr>
<tr>
<td>sailing</td>
<td>doc2</td>
</tr>
<tr>
<td>sailing</td>
<td>doc2</td>
</tr>
<tr>
<td>boats</td>
<td>doc2</td>
</tr>
<tr>
<td>sailing</td>
<td>doc3</td>
</tr>
<tr>
<td>east</td>
<td>doc3</td>
</tr>
<tr>
<td>coast</td>
<td>doc3</td>
</tr>
<tr>
<td>sailing</td>
<td>doc4</td>
</tr>
<tr>
<td>boats</td>
<td>doc5</td>
</tr>
</tbody>
</table>
CREATE VIEW City_Nationality AS
SELECT DISJOINT City, Nationality
FROM person | DISJOINT(Nationality);
### Motivation

A toy database is created to illustrate the concepts.

<table>
<thead>
<tr>
<th>Nationality</th>
<th>City</th>
</tr>
</thead>
<tbody>
<tr>
<td>German</td>
<td>London</td>
</tr>
<tr>
<td>Irish</td>
<td>London</td>
</tr>
<tr>
<td>German</td>
<td>Dortmund</td>
</tr>
<tr>
<td>German</td>
<td>Hamburg</td>
</tr>
<tr>
<td>Egyptian</td>
<td>London</td>
</tr>
<tr>
<td>Polish</td>
<td>London</td>
</tr>
<tr>
<td>Chinese</td>
<td>London</td>
</tr>
</tbody>
</table>

```sql
-- PSQL
CREATE VIEW Nationality_City AS
SELECT DISJOINT City, Nationality
FROM person | DISJOINT(City);
```
# PRA

Bayes <estAssumption>  
[<evidenceKey>](<prae>)

```sql
-- PSQL
SELECT ... FROM ... WHERE
ASSUMPTION <estAssumption>
EVIDENCE KEY <evidenceKey>
```

evidence = Project estAssumption[i1..in](a)

\[
T := T_a
\]

\[
P(\tau) = \begin{cases} 
\frac{P_a(\tau)}{P_{evidence}(\tau[i_1..i_n])} \\
\frac{\log P_a(\tau)}{\log P_{evidence}(\tau[i_1..i_n])} 
\end{cases}
\]

In 2003 (early Bayes), Bayes result was distinct; in 2004, revision led to \( T := T_a \).
Motivation: “Seamlessly” integrated DB+IR technology

- Support document/context and tuple retrieval
- Support free-text and semantic/data retrieval
- Support the flexible modelling of ALL retrieval models
- Support the high-level (abstract) modelling of general and specific retrieval tasks (ad-hoc retrieval, classification, summarisation, structured document retrieval, hypertext retrieval, multimedia retrieval, ...)
- Support text, XML, and SQL
Retrieval Models: Notation Notation ([Roelleke et al., 2006]):

- $n_D(t, x)$: number of Documents ... 
- $n_L(t, x)$: number of Locations ... 
- $P_D(t, x) := \frac{n_D(t, x)}{N_D(x)}$: document-based term probability 
- $P_L(t, x) := \frac{n_L(t, x)}{N_L(x)}$: location-based term probability 

...
Retrieval Models: RSV’s

\[ RSV_{\text{TF-IDF}}(d, q) := \sum_{t \in q} tf(t, d) \cdot \text{idf}(t, c) \quad (1) \]

\[ RSV_{\text{BIR}}(d, q) := \sum_{t \in d \cap q} \log \frac{P(t|r)}{P(t|\bar{r})} \quad (2) \]

\[ P(q|d) := \prod_{t \in q} [\delta \cdot P(t|d) + (1 - \delta) \cdot P(t|c)] \quad (3) \]
Retrieval Models: Relationships, Rewritings

\[ RSV_{BIR}(d, q) := \sum_{t \in d \cap q} idf(t, \bar{r}) - idf(t, r) \]  \hspace{1cm} (4)

\[ RSV_{LM}(d, q) := \log \frac{P(q|d)}{\prod_{t \in q}(1 - \delta) \cdot P(t|c)} \]  \hspace{1cm} (5)

\[ = \sum_{t \in q} \log \left[ 1 + \frac{\delta \cdot P(t|d)}{(1 - \delta) \cdot P(t|c)} \right] \]  \hspace{1cm} (6)

eqn 4: BIR and IDF:
[Robertson, 2004, de Vries and Roelleke, 2005] eqn 6: LM:
[Hiemstra, 2000]
Investigating the relationships important for “good” design of probabilistic relational modelling.
CREATE VIEW idf AS
    SELECT Term FROM coll
    ASSUMPTION MAX INFORMATIVE
    EVIDENCE KEY ();

CREATE VIEW wQuery AS
    SELECT Term, QueryId FROM Query, idf
    WHERE Query.Terms = idf.Terms;

CREATE VIEW norm_wQuery AS
    SELECT Term, QueryId FROM wQuery
    EVIDENCE KEY (QueryId);
1  
--- retrieve documents

2  CREATE VIEW std_tf_idf_retrieve AS

3     SELECT DISJOINT DocId, QueryId

4     FROM norm_wQuery, tf

5     WHERE norm_wQuery.Term = tf.Term;
# tf := P(t|d)
\[
{\text{tf}} = \text{Project DISJOINT (}
\begin{align*}
&\text{Bayes DISJOINT}[$2$](\text{coll}) ;
\end{align*}
\)]

# idf(t,c) :=
\[
# -\log P(t|c) / \max_{\text{idf}(c)}
\]
# max_idf(c)
\[
{\text{idf}} = \text{Bayes MAX_IDF[]} (\text{Project}[$1$](\text{coll})) ;
\]

-- P(t occurs | d)
\[
\text{CREATE VIEW tf AS SELECT DISJOINT Term, DocId FROM coll | DISJOINT (DocId)} ;
\]

-- P(t informs | c)
\[
\text{CREATE VIEW idf AS SELECT Term FROM coll ASSUMPTION MAX_IDF EVIDENCE KEY ()} ;
\]
CREATE VIEW idf_c AS
SELECT Term FROM coll
ASSUMPTION MAX_IDF
EVIDENCE KEY ();

CREATE VIEW idf_r AS
SELECT Term FROM relColl
ASSUMPTION MAX_IDF
EVIDENCE KEY ();
CREATE VIEW docModel AS
SELECT Term, DocId FROM lambda1, p_t_d;

CREATE VIEW collModel AS
SELECT Term, DocId FROM lambda2, p_t_c, retrieved;

-- combine document and collection models
CREATE VIEW lm1_mix AS
docModel UNION DISJOINT collModel;

-- retrieve documents
CREATE VIEW lm1_retrieve AS
SELECT SUM(LOG DocId, QueryId
FROM Query, lm1_mix
WHERE Query.Term = lm1_mix.Term;
Implementation

- Retrieval of $P(t|c)$ probabilities in $O(1)$ (special index)
- Incremental update facility
- $P(t|d)$ probabilities/views are materialised off-line; future research
- Difficulty: System A runs SQL, System B runs PSQL: How to compare?
- top-k and early-response processing
Summary

- Probabilistic DB, IR-on-DB, PRA, Retrieval Models
- The magnificent five (Select, Project, Join/Multiply, Unite, Subtract): describe probability *AGGREGATION*
- The relational Bayes: describe probability *ESTIMATION*
- DB+IR requires “relaxed” probability theory
  - *idf*-based (“informativeness”) probabilities
  - here and there, relax the boundaries of probabilistic modelling and rethink the genuine formulation of IR models
- Scalability: $O(1)$ retrieval of Bayes tuples for $P(t|c)$, given Bayes-oriented indexing structure
Outlook

- Database/tuple ranking: modelling and application of retrieval models to SELECT-FROM-WHERE; entropy-based ranking
- Design and verification of probabilistic logical programs
- Optimisation: Semantic, algebraic, and processing (Hengzhi Wu)
- High-level languages: RDF-SPARQL → PSQL/PRA (Hany Azzam)
- POLIS: Probabilistic Object-oriented logic for information summarisation (Fred Forst)
- POLAR: Probabilistic Object-oriented annotation-based retrieval (Ingo Frommholz)
Introduction
Background
The Relational Bayes Modelling Retrieval Models
Implementation
Summary and Outlook

Summary
Outlook

Dbxplorer: A system for keyword-based search over relational databases.
In ICDE, pages 5–16.

The management of probabilistic data.

Fuzzy queries and relational databases.

The theory of probabilistic databases.

Probabilistic ranking of database query results.
In VLDB, pages 888–899.

Probabilistic information retrieval approach for ranking of database query results.

A parameterised search system.
In ECIR, pages 4–15.
Using probabilistic models of document retrieval without relevance information.


Efficient query evaluation on probabilistic databases.

Relevance information: A loss of entropy but a gain for idf?
In ACM SIGIR, Salvador, Brazil.

The texture benchmark: Measuring performance of text queries on a relational dbms.
In VLDB, pages 313–324.

A probabilistic framework for vague queries and imprecise information in databases.

Dolores: A system for logic-based retrieval of multimedia objects.
In [Croft et al., 1998], pages 257–265.

A probabilistic NF2 relational algebra for integrated information retrieval and database systems.

A probabilistic relational algebra for the integration of information retrieval and database systems.


A probabilistic justification for using tf.idf term weighting in information retrieval.

Discover: Keyword search in relational databases.
In VLDB, pages 670–681.

Intelligent hypermedia retrieval.
Springer-Verlag Group (Physica-Verlag).

An extended relational database model for uncertain and imprecise information.

Text retrieval and the relational model.

Vague: A user interface to relational databases that permits vague queries.

Accommodating imprecision in database systems: Issues and solutions.

A straightforward NF2 relational interface with applications in information retrieval.

A language modeling approach to information retrieval.
In [Croft et al., 1998], pages 275–281.


The accessibility dimension for structured document retrieval.
In *Proceedings of the BCS-IRSG European Conference on Information Retrieval (ECIR), Glasgow*.

A general matrix framework for modelling information retrieval.
*Journal on Information Processing & Management (IP&M), Special Issue on Theory in Information Retrieval*, 42(1).

A parallel derivation of probabilistic information retrieval models.
In *ACM SIGIR*.

Natürlichsprachiger zugang zu datenbanken?

Data structures for an integrated database management and information retrieval system.
In *Proceedings of the 8th International Conference on Very Large Data Bases*, pages 197–207, Los Altos, California. Morgan Kaufman.

The relational model with relation-valued attributes.

Context-specific inverse document frequency for structured document retrieval.
In *European Conference on Information Retrieval (ECIR), London*. 
Poster.