Introduction
IR and Databases
The Logic View

Retrieval

- DB: given query \( q \), find objects \( o \) with \( o \rightarrow q \)
- IR: given query \( q \), find documents \( d \) with high values of \( P(d \rightarrow q) \)
- DB is a special case of IR! (in a certain sense)

This tutorial: Focusing on the logic view

- Inference
- Vague predicates
- Query language expressiveness
Inference

IR with the Relational Model
The Probabilistic Relational Model
Interpretation of probabilistic weights
Extensions
  - Disjoint events
  - Relational Bayes
  - Probabilistic rules
Relational Model

Projection

<table>
<thead>
<tr>
<th>DOCNO</th>
<th>TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ir</td>
</tr>
<tr>
<td>1</td>
<td>db</td>
</tr>
<tr>
<td>2</td>
<td>ir</td>
</tr>
<tr>
<td>3</td>
<td>db</td>
</tr>
<tr>
<td>3</td>
<td>oop</td>
</tr>
<tr>
<td>4</td>
<td>ir</td>
</tr>
<tr>
<td>4</td>
<td>ai</td>
</tr>
<tr>
<td>5</td>
<td>db</td>
</tr>
<tr>
<td>5</td>
<td>oop</td>
</tr>
</tbody>
</table>

**Projection:** what is the collection about?

\[
\text{topic}(T) :- \text{index}(D,T).
\]
Relational Model
Selection

<table>
<thead>
<tr>
<th>DOCNO</th>
<th>TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ir</td>
</tr>
<tr>
<td>1</td>
<td>db</td>
</tr>
<tr>
<td>2</td>
<td>ir</td>
</tr>
<tr>
<td>3</td>
<td>db</td>
</tr>
<tr>
<td>3</td>
<td>oop</td>
</tr>
<tr>
<td>4</td>
<td>ir</td>
</tr>
<tr>
<td>4</td>
<td>ai</td>
</tr>
<tr>
<td>5</td>
<td>db</td>
</tr>
<tr>
<td>5</td>
<td>oop</td>
</tr>
</tbody>
</table>

**Selection**: which documents are about IR?

aboutir(D) :- index(D,ir).
### Relational Model

#### Join

<table>
<thead>
<tr>
<th>index</th>
<th>DOCNO</th>
<th>TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>ir</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>db</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>ir</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>db</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>oop</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>ir</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>ai</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>db</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>oop</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>author</th>
<th>DOCNO</th>
<th>NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>smith</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>miller</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>johnson</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>firefly</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>firefly</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>bates</td>
</tr>
</tbody>
</table>

**Join**: who writes about IR?

irauthor(A):- index(D,ir) & author(D,A).
Relational Model

Union

<table>
<thead>
<tr>
<th>DOCNO</th>
<th>TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ir</td>
</tr>
<tr>
<td>1</td>
<td>db</td>
</tr>
<tr>
<td>2</td>
<td>ir</td>
</tr>
<tr>
<td>3</td>
<td>db</td>
</tr>
<tr>
<td>3</td>
<td>oop</td>
</tr>
<tr>
<td>4</td>
<td>ir</td>
</tr>
<tr>
<td>4</td>
<td>ai</td>
</tr>
<tr>
<td>5</td>
<td>db</td>
</tr>
<tr>
<td>5</td>
<td>oop</td>
</tr>
</tbody>
</table>

**index**

<table>
<thead>
<tr>
<th>DOCNO</th>
<th>TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ir</td>
</tr>
<tr>
<td>1</td>
<td>db</td>
</tr>
<tr>
<td>2</td>
<td>ir</td>
</tr>
<tr>
<td>3</td>
<td>db</td>
</tr>
<tr>
<td>3</td>
<td>oop</td>
</tr>
<tr>
<td>4</td>
<td>ir</td>
</tr>
<tr>
<td>4</td>
<td>ai</td>
</tr>
<tr>
<td>5</td>
<td>db</td>
</tr>
<tr>
<td>5</td>
<td>oop</td>
</tr>
</tbody>
</table>

**Union:** which documents are about IR or DB?

irordb(D) :- index(D,ir).

irordb(D) :- index(D,db).
Relational Model

Difference

<table>
<thead>
<tr>
<th>DOCNO</th>
<th>TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ir</td>
</tr>
<tr>
<td>1</td>
<td>db</td>
</tr>
<tr>
<td>2</td>
<td>ir</td>
</tr>
<tr>
<td>3</td>
<td>db</td>
</tr>
<tr>
<td>3</td>
<td>oop</td>
</tr>
<tr>
<td>4</td>
<td>ir</td>
</tr>
<tr>
<td>4</td>
<td>ai</td>
</tr>
<tr>
<td>5</td>
<td>db</td>
</tr>
<tr>
<td>5</td>
<td>oop</td>
</tr>
</tbody>
</table>

\[ \text{Difference: which documents are about IR, but not DB?} \]

\[ \text{irnotdb(D) :} \text{- index(D,ir) \& not(index(D,db)).} \]
Which documents are about DB?
aboutdb(D) :- index(D,db).

Which documents are about IR and DB?
aboutirdb(D) :- index(D,ir) & index(D,db).
Extensional vs. intensional semantics

<table>
<thead>
<tr>
<th>β</th>
<th>DOC</th>
<th>TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>d1</td>
<td>ir</td>
</tr>
<tr>
<td>0.5</td>
<td>d1</td>
<td>db</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>β</th>
<th>S</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7</td>
<td>d2</td>
<td>d1</td>
</tr>
</tbody>
</table>

about(D,T) :- docTerm(D,T).
about(D,T) :- link(D,D1) & about(D1,T)
q(D) :- about(D,ir) & about(D,db).

**extensional semantics:**
weight of derived fact as function of weights of subgoals
\[ P(q(d2)) = P(\text{about}(d2,\text{ir})) \cdot P(\text{about}(d2,\text{db})) = (0.7 \cdot 0.9) \cdot (0.7 \cdot 0.5) \]

**Problem**
“improper treatment of correlated sources of evidence” [Pearl 88]
→ extensional semantics only correct for tree-shaped inference structures
**Intensional semantics**

weight of derived fact as function of weights of underlying ground facts

**Method:** Event keys and event expressions

docterm

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>DOC</th>
<th>TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\beta)</td>
<td>(\kappa)</td>
<td>(dT(d1,ir))</td>
<td>(d1)</td>
</tr>
<tr>
<td>0.9</td>
<td>(dT(d1,db))</td>
<td>(d1)</td>
<td>(db)</td>
</tr>
</tbody>
</table>

?- docTerm(D,ir) & docTerm(D,db).
gives
\[d1 \ [dT(d1,ir) \& dT(d1,db)]\]

\[0.9 \cdot 0.5 = 0.45\]
Event keys and event expressions

docterm

<table>
<thead>
<tr>
<th>β</th>
<th>κ</th>
<th>DOC</th>
<th>TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>dT(d1,ir)</td>
<td>d1</td>
<td>ir</td>
</tr>
<tr>
<td>0.5</td>
<td>dT(d1,db)</td>
<td>d1</td>
<td>db</td>
</tr>
</tbody>
</table>

link

<table>
<thead>
<tr>
<th>β</th>
<th>κ</th>
<th>S</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7</td>
<td>l(d2,d1)</td>
<td>d2</td>
<td>d1</td>
</tr>
</tbody>
</table>

about(D,T) :- docterm(D,T).
about(D,T) :- link(D,D1) & about(D1,T)
?- about(D,ir) & about(D,db).

gives

\[d1 \ [dT(d1,ir) \land dT(d1,db)] \quad 0.9 \cdot 0.5 = 0.45\]
\[d2 \ [l(d2,d1) \land dT(d1,ir) \land l(d2,d1) \land dT(d1,db)] \quad 0.7 \cdot 0.9 \cdot 0.5 = 0.315\]
Recursion

about(D,T) :- docTerm(D,T).
about(D,T) :- link(D,D1) & about(D1,T).

?- about(D,ir)
d1 [dT(d1,ir)  |  l(d1,d2) & l(d2,d3) & l(d3,d1) &
          dT(d1,ir)  |  ...]  0.900
  d3 [l(d3,d1) & dT(d1,ir)]  0.720
  d2 [l(d2,d3) & l(d3,d1) & dT(d1,ir)]  0.288

?- about(D,ir) & about(D,db)
d1 [dT(d1,ir) & dT(d1,db)]  0.450
  d3 [l(d3,d1) & dT(d1,ir) & l(d3,d1) & dT(d1,db)]  0.360
  d2 [l(d2,d3) & l(d3,d1) & dT(d1,ir) & dT(d1,db)]  0.144
Computation of probabilities for event expressions

1. transformation of expression into disjunctive normal form
2. application of sieve formula:
   - simple case of 2 conjuncts: \( P(a \lor b) = P(a) + P(b) - P(a \land b) \)
   - general case:
     \( c_i \) – conjunct of event keys

\[
P(c_1 \lor \ldots \lor c_n) = 
\sum_{i=1}^{n} (-1)^{i-1} \sum_{1 \leq j_1 < \ldots < j_i \leq n} P(c_{j_1} \land \ldots \land c_{j_i}).
\]

- \( \leadsto \) exponential complexity
- \( \leadsto \) use only when necessary for correctness
- see [Dalvi & Suciu 07]
Possible worlds semantics

0.9 \text{docTerm}(d1,ir).

P(W_1) = 0.9: \{\text{docTerm}(d1,ir)\}

P(W_2) = 0.1: \{\}
0.6 docTerm(d1,ir). 0.5 docTerm(d1,db).

Possible interpretations:

$l_1$: $P(W_1) = 0.3$: \{docTerm(d1,ir)\}
    $P(W_2) = 0.3$: \{docTerm(d1,ir), docTerm(d1,db)\}
    $P(W_3) = 0.2$: \{docTerm(d1,db)\}
    $P(W_4) = 0.2$: \{

$l_2$: $P(W_1) = 0.5$: \{docTerm(d1,ir)\}
    $P(W_2) = 0.1$: \{docTerm(d1,ir), docTerm(d1,db)\}
    $P(W_3) = 0.4$: \{docTerm(d1,db)\}

$l_3$: $P(W_1) = 0.1$: \{docTerm(d1,ir)\}
    $P(W_2) = 0.5$: \{docTerm(d1,ir), docTerm(d1,db)\}
    $P(W_3) = 0.4$: \{

probabilistic logic:
$0.1 \leq P(\text{docTerm}(d1, ir) \& \text{docTerm}(d1, db)) \leq 0.5$

probabilistic Datalog with independence assumptions:
$P(\text{docTerm}(d1, ir) \& \text{docTerm}(d1, db)) = 0.3$
Disjoint events

<table>
<thead>
<tr>
<th>$\beta$</th>
<th>City</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7</td>
<td>Paris</td>
<td>France</td>
</tr>
<tr>
<td>0.2</td>
<td>Paris</td>
<td>Texas</td>
</tr>
<tr>
<td>0.1</td>
<td>Paris</td>
<td>Idaho</td>
</tr>
</tbody>
</table>

**Interpretation:**

$P(W_1) = 0.7: \{\text{cityState(paris, france)}\}$

$P(W_2) = 0.2: \{\text{cityState(paris, texas)}\}$

$P(W_3) = 0.1: \{\text{cityState(paris, idaho)}\}$
Relational Bayes

[Roelleke et al. 07]

Role of the relational Bayes: Generation of a probabilistic database

Diagram:

- Non-probabilistic database
- Probabilistic database

Arrow from Non-probabilistic database to Probabilistic database labeled Bayes
Relational Bayes
Example: $P(\text{Nationality} \mid \text{City})$

<table>
<thead>
<tr>
<th>nationality_and_city</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Nationality</td>
<td>City</td>
<td></td>
</tr>
<tr>
<td>&quot;British&quot;</td>
<td>&quot;London&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;British&quot;</td>
<td>&quot;London&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;Scottish&quot;</td>
<td>&quot;London&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;French&quot;</td>
<td>&quot;London&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;German&quot;</td>
<td>&quot;Hamburg&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;German&quot;</td>
<td>&quot;Hamburg&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;Danish&quot;</td>
<td>&quot;Hamburg&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;British&quot;</td>
<td>&quot;Hamburg&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;German&quot;</td>
<td>&quot;Dortmund&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;German&quot;</td>
<td>&quot;Dortmund&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;Turkish&quot;</td>
<td>&quot;Dortmund&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;Scottish&quot;</td>
<td>&quot;Glasgow&quot;</td>
<td></td>
</tr>
</tbody>
</table>

| nationality_city |  |          |          |
|------------------| |----------|----------|
| $P(\text{Nationality} \mid \text{City})$ | Nationality | City |
| 0.600            | "British"   | "London"|
| 0.200            | "Scottish"  | "London"|
| 0.200            | "French"    | "London"|
| 0.500            | "German"    | "Hamburg"|
| 0.250            | "Danish"    | "Hamburg"|
| 0.250            | "British"   | "Hamburg"|
| 0.667            | "German"    | "Dortmund"|
| 0.333            | "Turkish"   | "Dortmund"|
| 1.000            | "Scottish"  | "Glasgow"|

# $P(\text{Nationality} \mid \text{City})$:

1. `nationality_city SUM(Nat, City) :-
2. nationality_and_city (Nat, City) | (City);`
Relational Bayes

Example: $P(t|d)$

<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>boats</td>
<td>doc2</td>
</tr>
<tr>
<td>sailing</td>
<td>doc2</td>
</tr>
<tr>
<td>east</td>
<td>doc3</td>
</tr>
<tr>
<td>coast</td>
<td>doc3</td>
</tr>
<tr>
<td>sailing</td>
<td>doc3</td>
</tr>
<tr>
<td>sailing</td>
<td>doc4</td>
</tr>
<tr>
<td>boats</td>
<td>doc5</td>
</tr>
</tbody>
</table>

| $P(t|d)$ | Term | DocId |
|---------|------|------|
| 0.50    | sailing | doc1 |
| 0.50    | boats   | doc1 |
| 0.33    | sailing | doc2 |
| 0.33    | boats   | doc2 |
| 0.33    | sailing | doc2 |
| 0.33    | east    | doc3 |
| 0.33    | coast   | doc3 |
| 0.33    | sailing | doc3 |
| 1.00    | sailing | doc4 |
| 1.00    | boats   | doc5 |

| $P(t|d)$ | Term  | DocId |
|---------|-------|------|
| 0.50    | sailing | doc1 |
| 0.50    | boats   | doc1 |
| 0.67    | sailing | doc2 |
| 0.33    | boats   | doc2 |
| 0.33    | east    | doc3 |
| 0.33    | coast   | doc3 |
| 0.33    | sailing | doc3 |
| 1.00    | sailing | doc4 |
| 1.00    | boats   | doc5 |
Probabilistic rules
Rules for deterministic facts:

0.7 likes-sports(X) :- man(X).
0.4 likes-sports(X) :- woman(X).
man(peter).

Interpretation:
\[ P(W_1) = 0.7: \{\text{man(peter)}, \text{likes-sports(peter)}\} \]
\[ P(W_2) = 0.3: \{\text{man(peter)}\} \]
Probabilistic rules
Rules for uncertain facts:

#    gender is disjoint on the first attribute
0.7  l-sports(X) :- gender(X,male).
0.4  l-sports(X) :- gender(X,female).
0.5  gender(X,male) :- human(X).
0.5  gender(X,female) :- human(X).
human(jo).

Interpretation:

\[ P(W_1) = 0.35: \{ \text{gender(jo,male)}, \text{l-sports(jo)} \} \]
\[ P(W_2) = 0.15: \{ \text{gender(jo,male)} \} \]
\[ P(W_3) = 0.20: \{ \text{gender(jo,female)}, \text{l-sports(jo)} \} \]
\[ P(W_4) = 0.30: \{ \text{gender(jo,female)} \} \]

?- l-sports(jo) \quad \quad P(W_1) + P(W_3) = 0.55
Probabilistic rules
Rules for independent events

\[
\text{sameauthor}(D_1,D_2) :- \text{author}(D_1,X) \land \text{author}(D_2,X).
\]

\[
0.5 \text{ link}(D_1,D_2) :- \text{refer}(D_1,D_2).
\]

\[
0.2 \text{ link}(D_1,D_2) :- \text{sameauthor}(D_1,D_2).
\]

?? \text{link}(D_1,D_2) :- \text{refer}(D_1,D_2) \land \text{sameauthor}(D_1,D_2).

\[
P(l|r), P(l|s) \rightarrow P(l|r \land s)\]
Rules for independent events
Modeling probabilistic inference networks

0.7 \text{link}(D1,D2) :- \text{refer}(D1,D2) \& \text{sameauthor}(D1,D2).
0.5 \text{link}(D1,D2) :- \text{refer}(D1,D2) \& \text{not(sameauthor}(D1,D2)).
0.2 \text{link}(D1,D2) :- \text{sameauthor}(D1,D2) \& \text{not(refer}(D1,D2)).

**Probabilistic inference networks**, rules define link matrix

\[ \text{refer} \quad \text{sameauthor} \]

\[ \text{link} \]
Vague Predicates

The Logical View on Vague Predicates
Vague Predicates in IR and Databases
Probabilistic Modeling of Vague Predicates
Propositional vs. Predicate Logic

- Current IR systems are based on proposition logic (query term present/absent in document)
- Similarity of values not considered
- but multimedia IR deals with similarity already
- \(\leadsto\) transition from propositional to predicate logic necessary
- \(\Rightarrow\) Probabilistic databases / Datalog are already based on predicate logic!
Vague Predicates in Probabilistic Datalog

[Fuhr & Roelleke 97] [Fuhr 00]

- Example: Shopping 45 inch LCD TV
- Vague predicates as built-in predicates:
  \[ X \approx Y \]
- Query:
  \[ \text{query}(D) :\neg \text{Category}(D, \text{tv}) \land \text{type}(D, \text{lcd}) \land \text{size}(D, X) \land \approx(X, 45) \]

<table>
<thead>
<tr>
<th>( \beta )</th>
<th>( X )</th>
<th>( Y )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7</td>
<td>42</td>
<td>45</td>
</tr>
<tr>
<td>0.8</td>
<td>43</td>
<td>45</td>
</tr>
<tr>
<td>0.9</td>
<td>44</td>
<td>45</td>
</tr>
<tr>
<td>1.0</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td>0.9</td>
<td>46</td>
<td>45</td>
</tr>
<tr>
<td>0.8</td>
<td>47</td>
<td>45</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Data types and vague predicates in IR

Data type: domain + (vague) predicates

- Language (multilingual documents) / (language-specific stemming)
- Person names / “his name sounds like Jones”
- Dates / “about a month ago”
- Amounts / “orders exceeding 1 Mio $”
- Technical measurements / “at room temperature”
- Chemical formulas
Vague Criteria in Fact Databases

"I am looking for a 45-inch LCD TV with
- wide viewing angle
- high contrast
- low price
- high user rating"

→ vague criteria are very frequent in end-user querying of fact databases

→ but no appropriate support in SQL

vague conditions → similar to fuzzy predicates
Probabilistic Modeling of Vague Predicates

[Fuhr 90]

- learn vague predicates from feedback data
- construct feature vector $\vec{x}(q_i, d_i)$ from query value $q_i$ and document value $d_i$ (e.g. relative difference)
- apply logistic regression
Expressiveness

Retrieval Rules, Joins, Aggregations and Restructuring
Expressiveness in XML Retrieval
Expressiveness
Formulating Retrieval Rules

about(D,T) :- docTerm(D,T).

consider document linking / anchor text
about(D,T) :- link(D1,D),about(D1,T).

consider term hierarchy
about(D,T) :- subconcept(T,T1) & about(D,T1).

field-specific term weighting
0.9 docTerm(D,T) :- occurs(D,T,title).
0.5 docTerm(D,T) :- occurs(D,T,body).
Expressiveness

Joins

IR authors:

\texttt{irauthor(N):- about(D,ir) \& author(D,N).}

Smith’s IR papers cited by Miller

?\texttt{- author(D,smith) \& about(D,ir) \&}
\texttt{ \& author(D1,miller) \& cites(D,D1).}
Who are the major IR authors?

<table>
<thead>
<tr>
<th>$\beta$</th>
<th>DNO</th>
<th>TERM</th>
<th>$\beta$</th>
<th>DNO</th>
<th>NAME</th>
<th>$\beta$</th>
<th>DNO</th>
<th>NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>1</td>
<td>ir</td>
<td>1</td>
<td>1</td>
<td>smith</td>
<td>0.98</td>
<td>1</td>
<td>smith</td>
</tr>
<tr>
<td>0.8</td>
<td>1</td>
<td>db</td>
<td>2</td>
<td>2</td>
<td>miller</td>
<td>0.6</td>
<td>2</td>
<td>miller</td>
</tr>
<tr>
<td>0.6</td>
<td>2</td>
<td>ir</td>
<td>3</td>
<td>3</td>
<td>smith</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>3</td>
<td>ir</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>3</td>
<td>ai</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

irauthor(A):- index(D,ir) & author(D,A).

Aggregation through projection!
Expressiveness
Aggregation (2)

Who are the major IR authors?

<table>
<thead>
<tr>
<th>$\beta$</th>
<th>DNO</th>
<th>TERM</th>
<th>author DNO</th>
<th>NAME</th>
<th>irauths</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>1</td>
<td>ir</td>
<td>1</td>
<td>smith</td>
<td>1.7</td>
</tr>
<tr>
<td>0.8</td>
<td>1</td>
<td>db</td>
<td>2</td>
<td>miller</td>
<td>0.6</td>
</tr>
<tr>
<td>0.6</td>
<td>2</td>
<td>ir</td>
<td>3</td>
<td>smith</td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>3</td>
<td>ir</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>3</td>
<td>ai</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Aggregation through summing:

\[ \text{iraauth}(D,A) : \neg \text{index}(D,\text{ir}) \land \text{author}(D,A). \]
\[ \text{iraauths SUM} \text{Name} : \neg \text{irdbauth}(\text{Doc,Name}) \lor (\text{Name}) \]
Expressiveness in XML Retrieval

[Fuhr & Lalmas 07]

Content Typing

Object Types

Data Types

Text only

Structure

Nested structure

Named fields

XPath

XQuery
XML structure: 1. Nested Structure

- XML document as hierarchical structure
- Retrieval of elements (subtrees)
- Typical query language does not allow for specification of structural constraints
- Relevance-oriented selection of answer elements: return the most specific relevant elements
XML structure: 2. Named Fields

- Reference to elements through field names only
- Context of elements is ignored (e.g. author of article vs. author of referenced paper)
- Post-Coordination may lead to false hits (e.g. author name – author affiliation)

Example: Dublin Core

```xml
  <dc:title>Generic Algebras</dc:title>
  <dc:creator>A. Smith (ESI), B. Miller (CMU)</dc:creator>
  <dc:subject>Orthogonal group, Symplectic group</dc:subject>
  <dc:date>2001-02-27</dc:date>
  <dc:format>application/postscript</dc:format>
  <dc:source>ESI preprints</dc:source>
  <dc:language>en</dc:language>
</oai_dc:dc>
```
XML structure: 3. XPath

/doc/document/chapter[about(./heading, XML) AND about(./section///*,syntax)]
XML structure: 3. XPath (cont’d)

- Full expressiveness for navigation through document tree (+links)
  - Parent/child, ancestor/descendant
  - Following/preceding, following-sibling, preceding-sibling
  - Attribute, namespace

- Selection of arbitrary elements/subtrees
  (but answer can be only a single element of the originating document)
XML structure: 4. XQuery

Higher expressiveness, especially for database-like applications:

- Joins (trees → graphs)
- Aggregations
- Constructors for restructuring results

Example: List each publisher and the average price of its books
FOR $p$ IN distinct(document("bib.xml")//publisher)
LET $a := \text{avg}(\text{document}("bib.xml")//\text{book}[\text{publisher} = $p$]/\text{price})$
RETURN
  <publisher>
  <name> $p$/text() </name>
  <avgprice> $a </avgprice>
  </publisher>
XML content typing

Content Typing

Object Types

Data Types

Text only

Structure

Nested structure

Named fields

XPath

XQuery
This text explains all about XML and IR.

XML Query Language XQL

Examples

Syntax

Now we describe the XQL syntax.

Example query

//chapter[about(., XML query language]
XML content typing: 2. Data Types

- Data type: domain + (vague) predicates (see above)
- Close relationship to XML Schema, but
  - XMLS supports syntactic type checking only
  - No support for vague predicates
Object types: Persons, Locations. Dates, ..... 

Pablo Picasso (October 25, 1881 - April 8, 1973) was a Spanish painter and sculptor. In Paris, Picasso entertained a distinguished coterie of friends in the Montmartre and Montparnasse quarters, including André Breton, Guillaume Apollinaire, and writer Gertrude Stein.

To which other artists did Picasso have close relationships? Did he ever visit the USA?

Named entity recognition methods allow for automatic markup of object types

Object types support increased precision
XML content typing
Tag semantics modelled as hierarchies

Object type hierarchies

Scientist
  ↓
  Physicist  Chemist

Artist
  ↓
  Poet  Actor  Singer

Role hierarchies

Creator
  ↓
  Author  Editor
XML content typing
Tag semantics modelled in OWL
Further Concepts
Further Concepts
4-valued (probabilistic) logics

Supported concepts
- conflicting knowledge
- open + closed world assumptions

Applications
- 4-valued probabilistic Datalog [Fuhr & Roelleke 98]
- POOL: Probabilistic Object-Oriented Logic [Lalmas et al. 02]
- POLAR: Retrieval with Annotations [Frommholz & Fuhr 06]
- POLIS: Information summarization [Forst et al. 07]
IR Systems vs. DBMS

Separation between IRS and IR application?
Towards an IRMS

Application — SQL — DBMS

Application — IR Query Language — IRMS

DB

Collection
Conclusion
Conclusion

Inference

- Probabilistic relational model supports integration of IR+DB
- Probabilistic Datalog as powerful inference mechanism
- Allows for formulating retrieval strategies as logical rules

Vague predicates

- Natural extension of IR methods to attribute values
- Vague predicates can be learned from feedback data
- Transition from propositional to predicate logic

Expressive query language

- Joins
- Aggregations
- (Re)structuring of results
The HySpirit software development kit provides a descriptive approach for modelling complex information retrieval tasks such as hypermedia and knowledge retrieval by combining database models, probability theory, logic and object oriented concepts for the representation of knowledge and its intrinsic uncertainty.

Layered Architecture

References

Don’t *program* search engines, *design* them

http://www.spinque.com/


References II

Fuhr, N.; Rölleke, T.
(1997).
A Probabilistic Relational Algebra for the Integration of Information Retrieval and Database Systems.

Fuhr, N.; Rölleke, T.
HySpirit – a Probabilistic Inference Engine for Hypermedia Retrieval in Large Databases.

Fuhr, N.
A Probabilistic Framework for Vague Queries and Imprecise Information in Databases.

Fuhr, N.
(2000).
Probabilistic Datalog: Implementing Logical Information Retrieval for Advanced Applications.
*Journal of the American Society for Information Science 51*(2), pages 95–110.
Lalmas, M.; Roelleke, T.; Fuhr, N.
(2002).
Intelligent Hypermedia Retrieval.

Pearl, J.
(1988).

Rölleke, T.; Wu, H.; Wang, J.; Azzam, H.
(2007).
Modelling retrieval models in a probabilistic relational algebra with a new operator: the relational Bayes.
The International Journal on Very Large Data Bases (VLDB) 17(1), pages 5–37.

Suciu, D.; Olteanu, D.; Ré, C.; Koch, C.
(2011).
Probabilistic Databases.