Hybrid In-Database Inference for Declarative Information Extraction

Daisy Zhe Wang, Michael J. Franklin, Minos Garofalakis\textsuperscript{2}, Joseph M. Hellerstein, Michael Wick\textsuperscript{3}

University of California, Berkeley
Technical University of Crete\textsuperscript{2}
University of Massachusetts, Amherst\textsuperscript{3}
Information Extraction (IE)

“We are pleased that today's agreement guarantees our corporation will maintain a significant and long term presence in the Big Apple," McGraw-Hill president Harold McGraw III said in a statement.

--- From New York Times April 24, 1997
“We are pleased that today's agreement guarantees our corporation will maintain a significant and long term presence in the Big Apple,” McGraw-Hill president Harold McGraw III said in a statement.

--- From New York Times April 24, 1997

Labels:
Person  Company  Location  Other
"We are pleased that today's agreement guarantees our corporation will maintain a significant and long term presence in the Big Apple," McGraw-Hill president Harold McGraw III said in a statement.

--- From New York Times April 24, 1997

Labels:
Person   Company   Location   Other
Standard IE Systems

Tokenization, Feature Extraction, Inference …

SELECT * 
FROM Entities

Extracted Entities

Statistical ML Packages

Relational DBMS

SELECT * 
FROM Entities

Extracted Entities

Statistical ML Packages

Relational DBMS
Problems

- Poor Performance
  - Exhaustive Batch Analysis
  - No Indexing, Parallelization, Caching, Optimization in Data Analysis in SML

- Information Loss
  - DB only store Top-1 Highest Probability Results
  - Prematurely discard Uncertainties and Probabilities
Probabilistic Database

Declarative IE
Probabilistic Database & Declarative IE
BayesStore-IE: Linear-Chain CRF + Viterbi [ICDE10, VLDB10]

- In-Database Linear-Chain CRF and Viterbi
  → As efficient as open-source CRF package

- Top-k-style Relational Queries over probabilistic IE output
  → Reduce False Negatives by 80%
  → Orders-of-Magnitude Speedup

Limited Model, Single Inference, Limited Queries
Example Queries

```
SELECT Top-k(D1.docID, D2.docID)
FROM Emails^P D1, Emails^P D2
WHERE D1.docID != D2.docID
  and D1.Label^P = D2.Label^P = 'person'
  and D1.token = D2.token

SELECT Marginal(D1.docID, D2.docID, exist)
FROM Emails^P D1, Emails^P D2
WHERE D1.docID != D2.docID
  and D1.Label^P = D2.Label^P = 'person'
  and D1.token = D2.token
```
Contributions

- Efficient In-database Implementation of General MCMC Inference Algorithms
- Study of Factors affect the Choice of Inference Algorithms
- Hybrid Inference Optimization chooses the appropriate inference at the data item level
Why Different Inference Algorithms

Machine Learning

- Types of Inference
  - Marginal
  - Top-k

- Ground Model Structures
  - Linear-chain
  - Tree-shaped
  - Cyclic
What Inference Algorithms Used

<table>
<thead>
<tr>
<th>Inference Algorithms</th>
<th>Top-k</th>
<th></th>
<th></th>
<th>Marginal</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Linear</td>
<td>Tree</td>
<td>Cyclic</td>
<td>Linear</td>
<td>Tree</td>
<td>Cyclic</td>
</tr>
<tr>
<td>Viterbi (Max-Product)</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sum-Product</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>MCMC Gibbss</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>MCMC-MH</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>
In-Database MCMC Implementation

- Iterative Implementation
- Set-oriented implementation
  - Window Functions
  - Array Data Types
  - UDF functions

- Query-Driven MCMC Optimizations
- Guidelines for In-database implementation of statistical methods
What Inference Algorithms Used

<table>
<thead>
<tr>
<th>Inference Algorithms</th>
<th>Top-k</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Linear</td>
<td>Tree</td>
<td>Cyclic</td>
<td>Linear</td>
<td>Tree</td>
<td>Cyclic</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Viterbi (Max-Product)</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sum-Product</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>MCMC-Gibbs</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MCMC-MH</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

Trade-off between Accuracy and Runtime
Sum-Product vs. MCMC-MH, Gibbs

![Graph comparing Sum-Product and MCMC-MH, Gibbs vs. Sum-Product in terms of average probability difference over execution time.](image)
Viterbi vs. MCMC-MH, Gibbs

MCMC-MH, Gibbs vs. Viterbi

- Viterbi
- MCMC-MH
- Gibbs

Computation Error (Number of Labels)

Execution Time (sec)
Heuristic Rules for Choosing Inference Algorithms

<table>
<thead>
<tr>
<th>Inference Algorithms</th>
<th>Top-k</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Linear</td>
<td>Tree</td>
<td>Cyclic</td>
<td>Linear</td>
<td>Tree</td>
<td>Cyclic</td>
</tr>
<tr>
<td>Viterbi (Max-Product)</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sum-Product</td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>MCMCGibbs</td>
<td>*</td>
<td>*</td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>MCMC-MH</td>
<td>*</td>
<td>*</td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>
## Heuristic Rules for Choosing Inference Algorithms

<table>
<thead>
<tr>
<th>Inference Algorithms</th>
<th>Top-k</th>
<th>Marginal</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Linear</td>
<td>Tree</td>
</tr>
<tr>
<td>Viterbi (Max-Product)</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>Sum-Product</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MCMC-Gibbs</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>MCMC-MH</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>
Heuristic Rules for Choosing Inference Algorithms

<table>
<thead>
<tr>
<th>Inference Algorithms</th>
<th>Top-k</th>
<th></th>
<th></th>
<th>Marginal</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Linear</td>
<td>Tree</td>
<td>Cyclic</td>
<td>Linear</td>
<td>Tree</td>
<td>Cyclic</td>
</tr>
<tr>
<td>Viterbi (Max-Product)</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sum-Product</td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MCMCGibbs</td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MCMC-MH</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Heuristic Rules for Choosing Inference Algorithms

<table>
<thead>
<tr>
<th>Inference Algorithms</th>
<th>Top-k</th>
<th>Marginal</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Linear</td>
<td>Tree</td>
</tr>
<tr>
<td>Viterbi (Max-Product)</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>Sum-Product</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MCMCGibbs</td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>MCMC-MH</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Factors affect Choice of Inference

Machine Learning

- Types of Inference
  - Marginal
  - Top-k

- Ground Model Structures
  - Linear-chain
  - Tree-shaped
  - Cyclic

ProbDB

Extraction Model

Query

Data
Different Extraction Models

Linear-Chain CRF

Skip-Chain CRF
Query Changes Model: Probabilistic Join

CEO,Bill,Gates,talked,about,E1,exist,assigned,by,Bill,Clinton,today
Different Data $\rightarrow$ Different Model

Bill Gates met with Bill

E2 exists with E1

Bill assigned by Bill Clinton today
BayesStore-IE Query Processing

Result

Inference Query

groundCRFs

Ground over Data

CRF*

Probabilistic SPJ Query

Extraction CRF Model
Hybrid Inference Optimization

1. Apply Query to Extraction CRF Model $\rightarrow$ CRF*
   For Each Data Item
2. Instantiate Model over Data $\rightarrow$ groundCRF
3. Route $groundCRF$ to $linearCRF, treeCRF, cyclicCRF$
4. If Inference is Marginal
   apply sum-product to ($linearCRFs + treeCRFs$)
   apply MCMC Gibbs to ($cyclicCRFs$)
Else If Inference is Top-k
   apply Viterbi to ($linearCRFs$)
   apply max-product to ($treeCRFs$)
   apply MCMC Gibbs to ($cyclicCRFs$)
5. Union Results
Marginal over Probabilistic Join

1. $\text{Join}(\text{token1}=\text{token2} & \text{label1}=\text{label2})$
2. model instantiation
3. $\text{CRF1}$ & $\text{CRF2}$
4. Sum-product
5. $\text{MCMC Gibbs}$

only 1 cross-edge (tree) more than 2 cross-edges (cyclic)
Marginal over Probabilistic Join

1. \text{Join}(\text{token1}=\text{token2} \& \text{label1}=\text{label2})

2. \text{model instantiation}

3. only 1 cross-edge (tree) more than 2 cross-edges (cyclic)

4. \text{Sum-product}

5. \text{Union}

General, but Approximate and Slow

\text{MCMC Gibbs}

\text{CRF1-2}

\text{TokenTbl1} \quad \text{TokenTbl2}

\text{CRF1} \quad \text{CRF2}
## Preliminary Results

<table>
<thead>
<tr>
<th>Data Corpora</th>
<th>Probabilistic Join</th>
<th>Skip-Chain CRF</th>
</tr>
</thead>
<tbody>
<tr>
<td>NYTimes</td>
<td>4.5x</td>
<td>5.0x</td>
</tr>
<tr>
<td>Twitter</td>
<td>2.6x</td>
<td>5.0x</td>
</tr>
<tr>
<td>DBLP</td>
<td>1.0x</td>
<td>1.0x</td>
</tr>
</tbody>
</table>
Conclusion

- Efficient In-database Implementation of General MCMC Inference Algorithms

- Extraction Model, Query, and Data affect the Choice of Inference Algorithms

- Hybrid Inference Optimization can achieve up to 5x speed up on Twitter, NYTimes datasets
Related Work

Future Work

- MAD Library with EMC
- Cost-based Optimizer (Accuracy-Run time)
- In-Database Learning Algorithms
- NLP Pipeline with Reference Reconciliation
Thank you! ... Questions?

Daisy Zhe Wang Homepage:
http://www.cs.berkeley.edu/~daisyw

BayesStore Project Page:
http://www.cs.berkeley.edu/~daisyw/BayesStore.html

MAD Library (open source): http://madlib.net