



# PROKB Web-Scale Probabilistic Knowledge Base

Yang Chen

`yang@cise.ufl.edu`

Computer and Information Science and Engineering  
University of Florida

Feb 22, 2012



# Outline

## Introduction

Introduction

Examples

## The PROBKB System

PROBKB Architecture

Markov Logic

Parallel Processing Using GraphLab and Datapath

Incremental MCMC

## References

References

## Knowledge bases–Introduction

- A *knowledge base* [ND10] is a special kind of database for knowledge management. A knowledge base provides a means for information to be collected, organized, shared, searched and utilized.
- A knowledge base helps machines understand humans, languages, and the world.





# Knowledge bases—Introduction



About 478,000,000 results (0.52 seconds)

## Database - Microsoft Research

[research.microsoft.com/en-us/group/db/](http://research.microsoft.com/en-us/group/db/)

Increasing the usefulness of **database** systems to both business users and individuals.

## Which US universities have the best faculty for studying databases ...

[www.quora.com/Which-US-universities-have-the-best-faculty-...](http://www.quora.com/Which-US-universities-have-the-best-faculty-...)

Answer 1 of 5: Below is an incomplete list of schools that have very strong **database groups** in the **US**. Please forgive me if I forgot to include your favorite school.

## Search FishBase

[www.fishbase.org/](http://www.fishbase.org/)

Mirrors: [fishbase.org](http://fishbase.org/) | [fishbase.us](http://fishbase.us) | [fishbase.de](http://fishbase.de) | [fishbase.fr](http://fishbase.fr) | [fishbase.se](http://fishbase.se) | [fishbase.tw](http://fishbase.tw) | [fishbase.cn](http://fishbase.cn) | [fishbase.gr](http://fishbase.gr) | English | Español | Português (Br, Pt) ...

## The Database Group at Georgia Tech

[www.cc.gatech.edu/computing/Database/](http://www.cc.gatech.edu/computing/Database/)

The **research** theme of the **Database Group** emphasizes the needs of engineering and science applications as the driving forces behind the development of new ...

## Gale - Home

[www.gale.cengage.com/](http://www.gale.cengage.com/)

Select **Search Type**, Product Catalog, Site (e.g. Customer Service), Cart Wish List Sign In My ... **United States** | Change Your Region ... Outside **U.S.** and Canada ...

## Influenza Research Database: an integrated bioinformatics resource ...

[www.ncbi.nlm.nih.gov/pubmed/22260278](http://www.ncbi.nlm.nih.gov/pubmed/22260278)

by RB Squires · 2012 · Cited by 2 · Related articles

Jan 29, 2012 – Influenza **Research Database**: an integrated bioinformatics ... Davis, CA, **USA Southeast Poultry Research Lab**, **US** Department of ... **databases**, computational algorithms, external **research groups**, and the scientific literature.

## South Asia - World Bank Database Shows Export Markets Are ...

[web.worldbank.org](http://web.worldbank.org) > ... > News & Events > What's New

May 24, 2012 – **Search** South Asia, All. Click here for **search results** ... and International Integration team of the World Bank's Development **Research Group**.

## Data & Research - New Database Reveals Pattern of Services ...

[econ.worldbank.org](http://econ.worldbank.org) > Data & Research

Jul 9, 2012 – **New Database**: Transportation and professional services are especially protected ... manager of trade **research** at the World Bank's Development **Research Group**. ... The **U.S.** and **EU** account for more than 60 percent of world services ... from Europe to **South East Asia**, but with surprisingly little empirical ...



About 439,000,000 results (0.32 seconds)

## Berkeley Database Group | Research in data management and ...

[db.cs.berkeley.edu/](http://db.cs.berkeley.edu/)

Berkeley has led **database** systems **research** for over a quarter century. As data has become increasingly central to computing and society, our **group's** charter ...

People · Projects · MADlib goes beta · New CS194-17 Programming ...

## | Database Research Group, University of Michigan, Ann Arbor

[www.eecs.umich.edu/db/](http://www.eecs.umich.edu/db/)

The **database group** at Michigan is focused on building the data management infrastructure for the twenty-first century.

## UW CSE Database Group

[db.cs.washington.edu/](http://db.cs.washington.edu/)

**Database Research Group**. Drumheller Fountain at UW, Mt. Rainier in background. The University of Washington's database group aims at broadening the focus ...

## Database Research Group @ Columbia

[www.cs.columbia.edu/databases/](http://www.cs.columbia.edu/databases/)

**Database Research Group** @ Columbia. Announcements. We hosted the Spring '08 North East DBIR Day on April 18, 2008. Please see the event homepage ...

## Database - Microsoft Research

[research.microsoft.com/en-us/group/db/](http://research.microsoft.com/en-us/group/db/)

Increasing the usefulness of **database** systems to both business users and individuals.

## Database Research Group - Research - IBM

[www.research.ibm.com/sc/scalabledb/](http://www.research.ibm.com/sc/scalabledb/)

The home page of the Parallel **Databases Group** at the IBM T.J. Watson **Research Center**, NY.

## Top Southeastern Colleges - College Admissions - About.com

[collegearrivals.about.com/](http://collegearrivals.about.com/) / Top Southeastern Colleges ...

by Allen Grove · in 254 Google+ circles · More by Allen Grove

... frequently top the rankings for the **southeastern United States**.



Agnes Scott  
College  
Clemson  
University

Diliff / Wikimedia Commons  
Blue Sun Photography /  
Flickr

Location: Decatur, Georgia  
Location: Clemson, South  
...



# Outline

## Introduction

Introduction

Examples

## The PROBKB System

PROBKB Architecture

Markov Logic

Parallel Processing Using GraphLab and Datapath

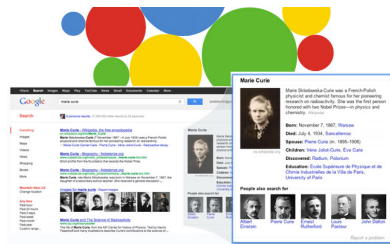
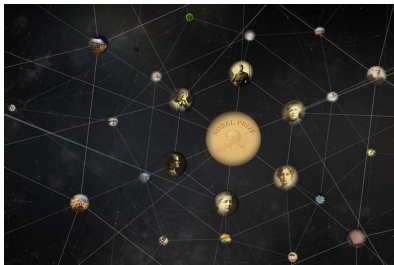
Incremental MCMC

## References

References

## Examples

### Google Knowledge Graph [Goo12]



### Demo



## Examples

### NELL

NELL [CBK<sup>+</sup>10] is a research project that attempts to create a computer system that learns over time to read the web.

Recently-Learned Facts [twitter](#) [Refresh](#)

instance	iteration	date learned	confidence
<a href="#">albert_ii</a> is a <a href="#">monarch</a>	623	13-aug-2012	99.9
<a href="#">kaseberg_creek</a> is a <a href="#">river</a>	620	01-aug-2012	99.1
<a href="#">senior_e_commerce_intelligence_consultant</a> is a <a href="#">profession</a>	622	09-aug-2012	98.4
<a href="#">deanna_anderson</a> is a <a href="#">journalist</a>	620	01-aug-2012	97.3
<a href="#">rose_city_comic_con</a> is a <a href="#">convention</a>	620	01-aug-2012	98.6
<a href="#">candidas</a> is a drug <a href="#">worked_on</a> by <a href="#">merck</a>	622	09-aug-2012	93.8
<a href="#">softball</a> is a sport <a href="#">with_fans_in</a> the country <a href="#">new_zealand_wellington</a>	623	13-aug-2012	100.0
<a href="#">mammals</a> is an animal that <a href="#">eats_berries</a>	622	09-aug-2012	100.0
<a href="#">wccu</a> is a <a href="#">TV_affiliate_of</a> the network <a href="#">fox</a>	625	21-aug-2012	100.0
<a href="#">cats</a> is an animal that can <a href="#">develop_respiratory_infection</a>	622	09-aug-2012	100.0

Note that NELL produces uncertain results. This is typical among automatic extraction systems and is our major motivation to develop a large-scale probabilistic knowledge base.

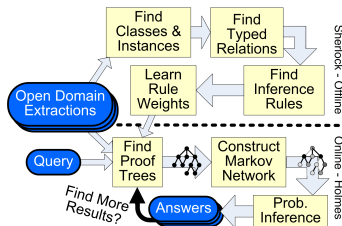
## Examples

### SHERLOCK-HOLMES

The

SHERLOCK-HOLMES [Sch11] is an open information extraction system consisting of two components:

- SHERLOCK [SEWD10], which learns inference rules offline, and
- HOLMES [SEW08], which uses inference rules to answer queries online.





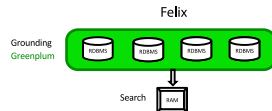


## Examples

### TUFFY-FELIX

The TUFFY-FELIX [NRDS11, NZRS11] system is an Markov logic network [RD06] implementation that does large-scale probabilistic inference using an RDBMS.

- A bottom-up approach to grounding using an RDBMS.
- A hybrid in-database grounding and in-memory inference architecture.
- Novel partitioning, loading, and parallel algorithms.
- Task decomposition to achieve web-scale.





# Outline

## Introduction

Introduction

Examples

## The PROBKB System

PROBKB Architecture

Markov Logic

Parallel Processing Using GraphLab and Datapath

Incremental MCMC

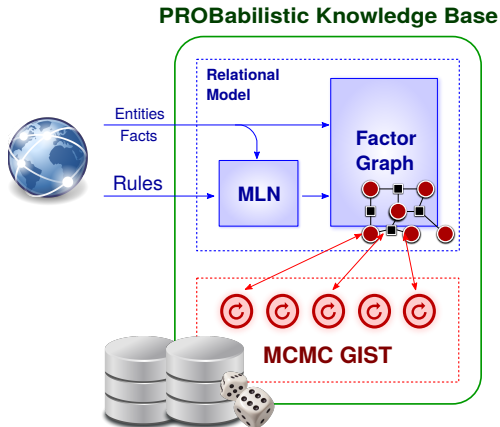
## References

References



# Architecture

- Extracted entities, facts, and rules stored as relational model.
- Efficient grounding via a few relational operators.
- Parallel MCMC inference implemented as GIST operations.
- Incremental inference: save computation by focusing on least convergent variables.





# Outline

## Introduction

Introduction

Examples

## The PROBKB System

PROBKB Architecture

Markov Logic

Parallel Processing Using GraphLab and Datapath

Incremental MCMC

## References

References



## Markov Logic–Probabilistic Inference Framework

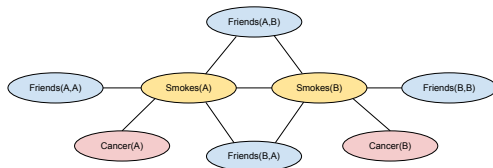
A *Markov logic network* (MLN) [RD06] is a set of formulae with weights. Together with a finite set of constants  $C = \{c_1, \dots, c_{|C|}\}$ , it defines a Markov network.

Weight	First-Order Logic
0.7	$\text{Fr}(x, y) \wedge \text{Fr}(y, z) \rightarrow \text{Fr}(x, z)$
1.5	$\text{Sm}(x) \rightarrow \text{Ca}(x)$
1.1	$\text{Fr}(x, y) \wedge \text{Sm}(x) \rightarrow \text{Sm}(y)$

A set of constants (entities, or objects)

**Table:** Example Markov logic network.

$$C = \{A, B\}.$$



**Figure:** Grounded Markov network.



# Markov Logic Inference

## Grounding

*Grounding* is the process of substituting constants into MLN clauses.

The result of grounding is a *factor graph* (or *Markov network*) from which we can infer marginal probabilities for individual facts.

## Key Challenges

- Time-consuming, especially if the numbers of rules and entites are large.
- Grounded network has an intractably large size, making inference tasks slow.



# Markov Logic Inference

## Scaling to the Web

- First-order Horn clauses:
  - Avoids the need to enumerate ground atoms.
  - Stored as *first-class* citizen in RDBMS, grounding expressed as a few `Join` s.
  - Easier to learn than general first-order clauses.
- Leveraging RDBMS query optimization techniques and possibly MPP frameworks (e.g. Greenplum).
- Ontology (typing): reducing the number of possible groundings and improving accuracy.



## Grounding

A single JOIN operation handles all rules of type

$$p(x : c_1, y : c_2) \leftarrow q(x : c_1, z : c_3), r(z : c_3, y : c_2)$$

```

SELECT DISTINCT mln.head AS head, mln.body1 AS body1,
    mln.body2 AS body2, r1.ent1 AS ent1, r1.ent2 AS
    ent2, r2.ent2 AS ent3
FROM relations r1, mln, relations r2, relations r3,
    instances i1, instances i2, instances i3
WHERE r1.pred = mln.head AND r2.pred = mln.body1 AND
    r3.pred = mln.body2
AND r1.ent1 = r2.ent1 AND r1.ent2 = r3.ent2 AND r2.ent2
    = r3.ent1
AND i1.ent = r1.ent1 AND i1.class = mln.class1
AND i2.ent = r1.ent2 AND i2.class = mln.class2
AND i3.ent = r2.ent2 AND i3.class = mln.class3
  
```





## Results

We can ground the whole SHERLOCK-HOLMES dataset the first two rounds in 10 minutes using PostgreSQL, while the state-of-the-art implementation MLN (TUFFY [NRDS11]) crashes during its grouding phase.

#relations	10,672
#rules	31,000
#constants	1.1M
#evidence	250,000
#queries	10,672
TUFFY	<b>Crash</b>
PROKB	10 min <sup>1</sup>

**Table:** Dataset statistics and performance.

---

<sup>1</sup>First two rounds.



# Inference

## Inference—Computing Probabilities

The Markov random field defines a probability distribution on all nodes in it:

$$P(\mathbf{X} = \mathbf{x}) = \frac{1}{Z} \exp \left( \sum_i w_i n_i(\mathbf{x}) \right),$$

where  $n_i$  is the number of ground clauses that satisfy clause  $i$  in the MLN and  $w_i$  is the weight of that clause.  $Z$  is the normalization constant, also called the *partition function*.



# MCMC-MH

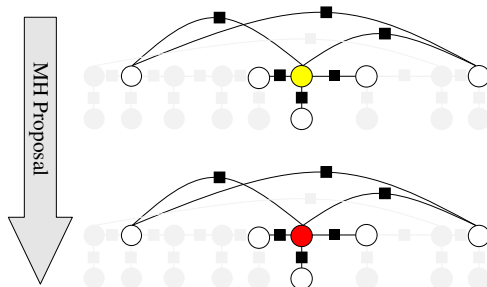
## Metropolis-Hastings (MH)

- Exact inference: intractable due to  $Z$ .
- MCMC-MH: efficient since  $Z$  cancels out.
- MCMC-MH: only changed factors need to be considered [WMM10]:

$$\begin{aligned}\frac{\pi(\mathbf{x})}{\pi(\mathbf{x}')} &= \frac{\frac{1}{Z} \prod_i \phi_i(\mathbf{x}_i)}{\frac{1}{Z} \prod_i \phi_i(\mathbf{x}'_i)} \\ &= \frac{\prod_{i \text{ having } x_{(k)}} \phi_i(x_{(k)}, \mathbf{x})}{\prod_{i \text{ having } x_{(k)}} \phi_i(x'_{(k)}, \mathbf{x})}.\end{aligned}$$



## MCMC-MH Efficiency





# Outline

## Introduction

- Introduction

- Examples

## The PROBKB System

- PROBKB Architecture

- Markov Logic

- Parallel Processing Using GraphLab and Datapath

- Incremental MCMC

## References

- References



# GraphLab

Data-Parallel

Graph-Parallel

## Map Reduce

Feature Extraction      Cross Validation  
Computing Sufficient Statistics

**GraphLab**  
Carnegie Mellon



**Graphical Models**

Gibbs Sampling  
Belief Propagation  
Variational Opt.

**Semi-Supervised Learning**

Label Propagation  
CoEM

**Collaborative Filtering**

Tensor Factorization

**Data-Mining**  
PageRank

Triangle Counting





# GraphLab Execution Model

---

## Algorithm 1 GraphLab Execution Model

---

**Input:** Data graph  $G = (V, E, D)$

**Input:** Initial vertex set  $\mathcal{T} = \{v_1, v_2, \dots\}$

**while**  $\mathcal{T}$  is not empty **do**

$v \leftarrow \text{RemoveNext}(\mathcal{T})$

$(\mathcal{T}', \mathcal{S}_v) \leftarrow f(v, \mathcal{S}_v)$

$\mathcal{T} \leftarrow \mathcal{T} \cup \mathcal{T}'$

**Output:** Modified data graph  $G = (V, E, D')$

---

- Vertexes schedule execution of their neighbors.
- Not applicable to general MCMC algorithms.



# Datapath GIST

## Generalized Iterable State Transforms (GIST)

- GIST Performs *transitions* upon a *state* until that state has converged to the desired result.

**Transition** MCMC Proposal function.

**State** Factor graph with its samples.

- A user-defined local scheduler allows general MCMC proposal implementation.
- The GIST state keeps track of the inference result.





## GraphLab Preliminary Results

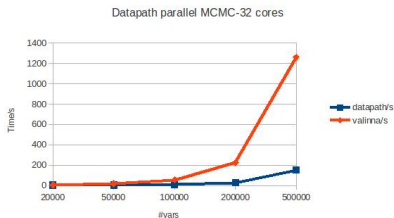
#samples	10	100	200	500	State-of-the-art
<b>IE</b>	0.2s	2s	4s	10.2s	25.216s
<b>ER</b>	90.8s	181.5s	373s	>600s	225s
<b>RC</b>	5.2s	52.7s	111.3s	297.8s	<b>Crashed</b>
SHERLOCK-600	1.2s	12.6s	28.3s	65.1s	55min

**Table:** GraphLab-based parallel inference vs the state-of-the-art.



# Datapath Preliminary Results

Figure: Inference over simulated factor graphs





# Outline

## Introduction

Introduction

Examples

## The PROBKB System

PROBKB Architecture

Markov Logic

Parallel Processing Using GraphLab and Datapath

Incremental MCMC

## References

References



# Incremental MCMC

## Goal

- EFBP [ND10] in a MCMC setting

## Incremental MCMC

The incremental maintenance algorithm adopts the Query-aware MCMC [WM11] technique:

- Assumption: not much new information is added to the knowledge base each time.
- Newly extracted knowledge serves as the query node.
- Maintaining samples for both old and new nodes.



# Incremental Maintenance of MCMC Samples

The incremental MCMC proposal function  $T$  employs the following steps:

1. Sample the variable index space according to some distribution  $p$  reflecting influence of new nodes and recent sample behaviors.
2. Sample the selected variable according to some distribution  $q$  over that variable's domain, leaving all other variables unchanged.

We adjust the distribution  $p$  so that it focuses on newly added variables.



## Conclusions

- PROBKB is a web-scale **PROB**abilistic **K**nowledge **B**ase with Markov logic network as the primary data model.
- All extractions, including entities and rules, are stored in RDBMS as relational model, allowing efficient grounding algorithms.
- In-database GIST operator allows parallel MCMC inference.
- Incremental MCMC focuses computation on most recently added variables, speeding up convergence.



# Questions?

# Thank you!



# Outline

## Introduction

- Introduction

- Examples

## The PROBKB System

- PROBKB Architecture

- Markov Logic

- Parallel Processing Using GraphLab and Datapath

- Incremental MCMC

## References

- References





## References I



A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E.R. Hruschka Jr, and T.M. Mitchell.

Toward an architecture for never-ending language learning.  
*In Proceedings of the Conference on Artificial Intelligence (AAAI)*, pages 1306–1313, 2010.



Google.

Introducing the knowledge graph: things, not strings, 2012.






A. Nath and P. Domingos.

Efficient belief propagation for utility maximization and repeated inference.

*In Proceedings of the 24th AAAI Conference on Artificial Intelligence*, 2010.



## References II

-  Feng Niu, Christopher Ré, AnHai Doan, and Jude W. Shavlik.  
Tuffy: Scaling up statistical inference in markov logic networks using an rdbms.  
*PVLDB*, 4(6):373–384, 2011.
-  F. Niu, C. Zhang, C. Ré, and J. Shavlik.  
Felix: Scaling Inference for Markov Logic with an Operator-based Approach.  
*ArXiv e-prints*, August 2011.
-  M. Richardson and P. Domingos.  
Markov logic networks.  
*Machine learning*, 62(1):107–136, 2006.



## References III



S. Schoenmackers.

*Inference Over the Web.*

PhD thesis, University of Washington, 2011.



S. Schoenmackers, O. Etzioni, and D.S. Weld.

Scaling textual inference to the web.

In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 79–88. Association for Computational Linguistics, 2008.



S. Schoenmackers, O. Etzioni, D.S. Weld, and J. Davis.

Learning first-order horn clauses from web text.

In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1088–1098.

Association for Computational Linguistics, 2010.





## References IV



Michael L Wick and Andrew McCallum.

Queryaware mcmc.

*In proceedings of the 25th Conference on Neural Information Processing Systems (NIPS)*, pages 2564–2572, 2011.



M. Wick, A. McCallum, and G. Miklau.

Scalable probabilistic databases with factor graphs and mcmc.

*Proceedings of the VLDB Endowment*, 3(1-2):794–804, 2010.