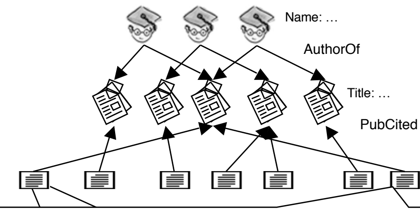
# Unknown Objects and BLOG

Brian Milch
MIT

IPAM Summer School
July 16, 2007

## Example 1: Bibliographies

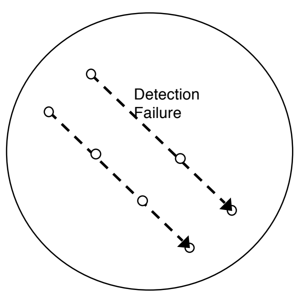



Russell, Stuart and Norvig, Peter. Articial Intelligence. Prentice-Hall, 1995.

S. Russel and P. Norvig (1995). Artificial Intelligence: A Modern Approach. Upper Saddle River, NJ: Prentice Hall.

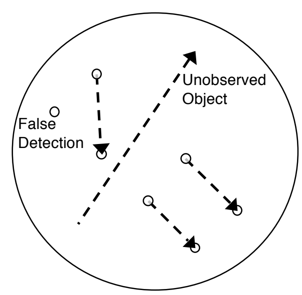
2

## Example 2: Aircraft Tracking



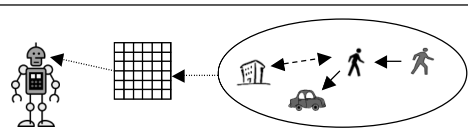

3

## Example 2: Aircraft Tracking




4

## Handling Unknown Objects



- Fundamental task: given observations, make inferences about initially unknown objects
- But most RPM languages assume set of objects is fixed and known (Herbrand models)
- Bayesian logic (BLOG) lifts this assumption

[Milch *et al.*, IJCAI 2005. See also MEBN: Laskey & da Costa, UAI 2005; Dynamical Grammars: Mjolsness & Yosiphon, AMAI to appear]

5

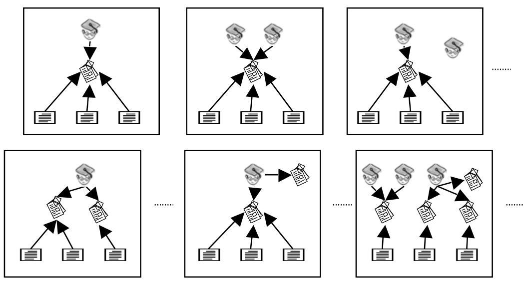
## Outline

- BLOG models with unknown objects
  - Syntax
  - Semantics
- Inference with unknown objects
  - Likelihood weighting
  - MCMC
- Applications
  - Citation matching
  - Multi-object tracking
- Alternative approaches

6

## Possible Worlds

(not showing attribute values)



How can we define a distribution over such outcomes?



## Generative Process

- Imagine process that constructs worlds using two kinds of steps
  - Add some objects to the world
  - Set the value of a function on a tuple of arguments



## BLOG Model for Citations

```
#Paper ~ NumPapersPrior();                  <- number statement

Title(p) ~ TitlePrior();
                                           part of skeleton:
                                           exhaustive list of distinct citations
guaranteed Citation Cit1, Cit2, Cit3, Cit4, Cit5, Cit6, Cit7;

PubCited(c) ~ Uniform({Paper p});          <- familiar syntax for
                                              reference uncertainty

Text(c) ~ NoisyCitationGrammar(Title(PubCited(c)));
```

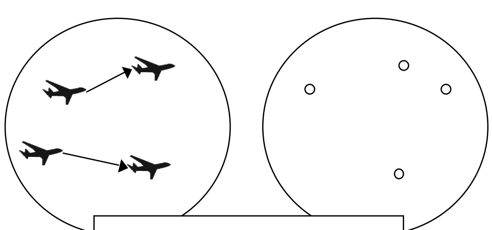


## Adding Authors

```
#Researcher ~ NumResearchersPrior();
Name(r) ~ NamePrior();
#Paper ~ NumPapersPrior();
FirstAuthor(p) ~ Uniform({Researcher r});
Title(p) ~ TitlePrior();
PubCited(c) ~ Uniform({Paper p});
Text(c) ~ NoisyCitationGrammar
        (Name(FirstAuthor(PubCited(c))), Title(PubCited(c)));
```



## Generative Process for Aircraft Tracking

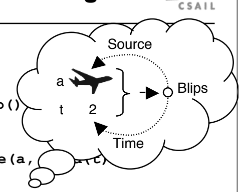


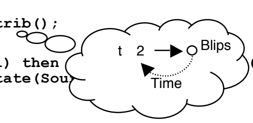
Existence of radar blips depends on existence and locations of aircraft



## BLOG Model for Aircraft Tracking

```
…
#Aircraft ~ NumAircraftDistrib()
State(a, t)
  if t = 0 then ~ InitState()
  else ~ StateTransition(State(a,
#Blip(Source = a, Time = t)
  ~ NumDetectionsDistrib(State(a, t));
#Blip(Time = t)
  ~ NumFalseAlarmsDistrib();
ApparentPos(r)
  if (Source(r) = null) then                    ()
  else ~ ObsDistrib(State(Sou
```



## Declarative Semantics

- What is the set of possible worlds?
- What is the probability distribution over worlds?

13

## What Exactly Are the Objects?

- Objects are tuples that encode generation history
  - Aircraft: (Aircraft, 1), (Aircraft, 2), …
  - Blips from (Aircraft, 2) at time 8:
    (Blip, (Source, (Aircraft, 2)), (Time, 8), 1)
    (Blip, (Source, (Aircraft, 2)), (Time, 8), 2)
    …
- Point: If we specify *how many* blips were generated by (Aircraft, 2) at time 8, there's no ambiguity about *which* blips were generated

14

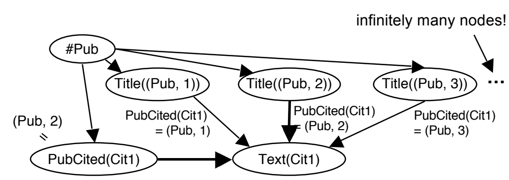
## Basic Random Variables (RVs)

- For each number statement and tuple of generating objects, have RV for number of objects generated
- For each function symbol and tuple of arguments, have RV for function value
- *Lemma*: Full instantiation of these RVs uniquely identifies a possible world

15

## Contingent Bayesian Network

- Each BLOG model defines contingent Bayesian network (CBN) over basic RVs
  - Edges active only under certain conditions




[Milch *et al.*, AI/Stats 2005] 16

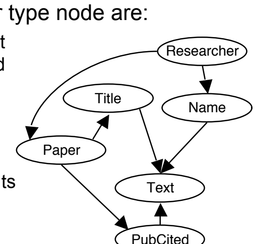
## Probability Distribution

- Through its CBN, BLOG model specifies:
  - Conditional distributions for basic RVs
  - Context-specific independence properties
    *e.g.*, Text(Cit1) indep of Title((Pub, 1))
    given PubCited(Cit1) = (Pub, 3)
- *Theorem*: Under certain conditions (analogous to BN acyclicity), every BLOG model defines unique distribution over possible worlds

[Milch *et al.*, IJCAI 2005] 17

## Symbol Graphs and Unknown Objects

- Symbol graph now contains not only random functions, but random types
- Parents of a function or type node are:
  - Functions and types that appear on the right hand side of dependency or number statements for this function/type
  - The types of this function/type's arguments or generating objects




18

## Outline

- BLOG models with unknown objects
  - Syntax
  - Semantics
- Inference with unknown objects
  - Likelihood weighting
  - MCMC
- Applications
  - Citation matching
  - Multi-object tracking
- Alternative approaches

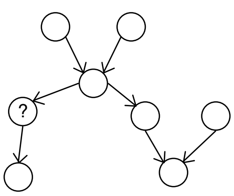
19

## Inference for BLOG

- Does infinite set of basic RVs prevent inference?
- No: Sampling algorithm only needs to instantiate finite set of relevant variables
- Algorithms:
  - Rejection sampling [Milch *et al.*, IJCAI 2005]
  - Guided likelihood weighting [Milch *et al.*, AI/Stats 2005]
- *Theorem*: For any well-formed BLOG model, these sampling algorithms converge to correct probability for any query, using finite time per sampling step

20

## Approximate Inference by Likelihood Weighting

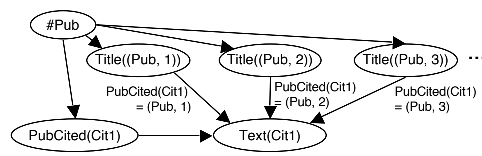



- Sample non-evidence nodes top-down
- Weight each sample by product of probabilities of evidence nodes given their parents
- Provably converges to correct posterior

21

## Application to BLOG

- Only need to sample ancestors of query and evidence nodes
- But until we condition on PubCited(Cit1), Text(Cit1) has infinitely many parents
- Solution: interleave sampling and relevance determination




22

## Likelihood Weighting for (Simplified) Citation Matching

| | Instantiation | Stack |
|---|---|---|
| Evidence:<br>✓Text(Cit1) = “foo”;<br>✓Text(Cit2) = “foob”;<br>Query:<br>✓#Paper | #Paper = 7<br>PubCited(Cit1) = (Paper, 3)<br>Title((Paper, 3)) = “Foo”<br>Text(Cit1) = “foo”<br>PubCited(Cit2) = (Paper, 3)<br>Text(Cit2) = “foob”<br>Weight: 1 x 0.8 x 0.2 | |

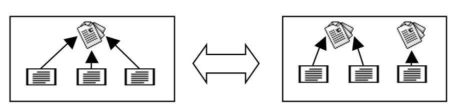
```
#Paper ~ NumPapersPrior();
Title(p) ~ TitlePrior();
PubCited(c) ~ Uniform({Paper p});
Text(c) ~ NoisyCitationGrammar(Title(PubCited(c));
```

More realistically: use MCMC

23

## MCMC for Citations

- Split-merge moves:



  - Propose titles and author names for affected publications based on citation strings
- Other moves change total number of publications

[Pasula *et al.*, NIPS 2002]

24

## MCMC States

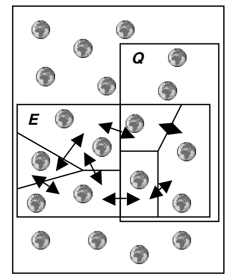
- Not complete instantiations!
  - No titles, author names for uncited publications
- States are partial instantiations of random variables

  #Pub = 100, PubCited(Cit1) = (Pub, 37), Title((Pub, 37)) = "Calculus"
  - Each state corresponds to an event: set of outcomes satisfying description

25

## MCMC over Events

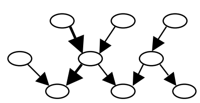
- Markov chain over events $\sigma$, with stationary distrib. proportional to $\boldsymbol{p}(\sigma)$
- *Theorem:* Fraction of visited events in $\boldsymbol{Q}$ converges to $\boldsymbol{p}(\boldsymbol{Q}|\boldsymbol{E})$ if:
  - Each $\sigma$ is either subset of $\boldsymbol{Q}$ or disjoint from $\boldsymbol{Q}$
  - Events form partition of $\boldsymbol{E}$




[Milch & Russell, UAI 2006] 26

## Computing Probabilities of Events

- Engine needs to compute $\boldsymbol{p}(\sigma') / \boldsymbol{p}(\sigma_{\boldsymbol{n}})$ efficiently (without summations)
- Use instantiations that include all active parents of the variables they instantiate



- Then probability is product of CPDs:

27

## Outline

- BLOG models with unknown objects
  - Syntax
  - Semantics
- Inference with unknown objects
  - Likelihood weighting
  - MCMC
- Applications
  - Citation matching
  - Multi-object tracking
- Alternative approaches

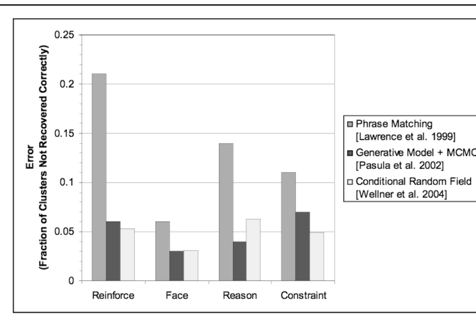
28

## Citation Matching

- Elaboration of generative model shown earlier
- Parameter estimation
  - Priors for names, titles, citation formats learned offline from labeled data
  - String corruption parameters learned with Monte Carlo EM
- Inference
  - MCMC with split-merge proposals
  - Guided by "canopies" of similar citations
  - Accuracy stabilizes after ~20 minutes

[Pasula *et al.*, NIPS 2002] 29

## Citation Matching Results




Four data sets of ~300-500 citations, referring to ~150-300 papers

30

## Cross-Citation Disambiguation

```
Wauchope, K. Eucalyptus: Integrating Natural Language
Input with a Graphical User Interface.  NRL Report
NRL/FR/5510-94-9711 (1994).
```

Is "Eucalyptus" part of the title, or is the author named K. Eucalyptus Wauchope?

```
Kenneth Wauchope (1994). Eucalyptus: Integrating
natural language input with a graphical user
interface.  NRL Report NRL/FR/5510-94-9711, Naval
Research Laboratory, Washington, DC, 39pp.
```

Second citation makes it clear how to parse the first one



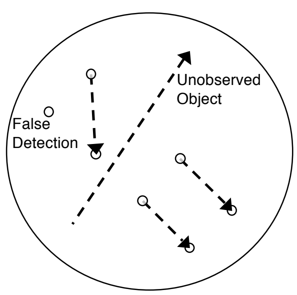
## Preliminary Experiments: Information Extraction

- P(citation text | title, author names) modeled with simple HMM
- For each paper: recover title, author surnames and given names
- Fraction whose attributes are recovered perfectly in last MCMC state:
  - among papers with one citation: 36.1%
  - among papers with multiple citations: 62.6%

Can use inferred knowledge for disambiguation



## Multi-Object Tracking






## State Estimation for "Aircraft"

```
#Aircraft ~ NumAircraftPrior();
State(a, t)
  if t = 0 then ~ InitState()
  else ~ StateTransition(State(a, Pred(t)));
#Blip(Source = a, Time = t)
  ~ NumDetectionsCPD(State(a, t));
#Blip(Time = t)
  ~ NumFalseAlarmsPrior();
ApparentPos(r)
  if (Source(r) = null) then ~ FalseAlarmDistrib()
  else ~ ObsCPD(State(Source(r), Time(r)));
```



## Aircraft Entering and Exiting

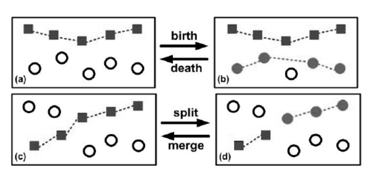
```
#Aircraft(EntryTime = t) ~ NumAircraftPrior();
Exits(a, t)
  if InFlight(a, t) then ~ Bernoulli(0.1);
InFlight(a, t)
  if t < EntryTime(a) then = false
  elseif t = EntryTime(a) then = true
  else = (InFlight(a, Pred(t)) & !Exits(a, Pred(t)));
State(a, t)
  if t = EntryTime(a) then ~ InitState()
  elseif InFlight(a, t) then
      ~ StateTransition(State(a, Pred(t)));
#Blip(Source = a, Time = t)
  if InFlight(a, t) then
      ~ NumDetectionsCPD(State(a, t));
```

…plus last two statements from previous slide



## MCMC for Aircraft Tracking

- Uses generative model from previous slide (although not with BLOG syntax)
- Examples of Metropolis-Hastings proposals:

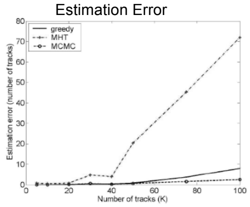



[Figures by Songhwai Oh]

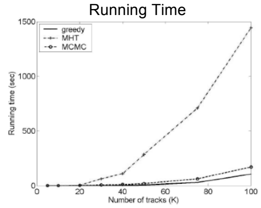
[Oh et al., CDC 2004]

## Aircraft Tracking Results

Estimation Error




Running Time




MCMC has smallest error, hardly degrades at all as tracks get dense

MCMC is nearly as fast as greedy algorithm; much faster than MHT

[Figures by Songhwai Oh]

[Oh et al., CDC 2004]



## BLOG Software

- Bayesian Logic inference engine available:

  http://people.csail.mit.edu/milch/blog



## Alternative Approaches

- Many alternative languages for representing RPMs
- Surveys:
  - ILP invited talk [Milch & Russell 2006]
  - Forthcoming book, *Statistical Relational Learning*, L. Getoor and B. Taskar, eds.



## RPMs Based on Directed Graphical Models (besides BLOG)

- BUGS language [Gilks et al., 1994]
  - Relational parent specs: height[father[i]]
  - Gibbs sampling engine available
- Probabilistic relational models (PRMs) [Koller & Pfeffer 1998; Friedman, Getoor, Koller & Pfeffer 1999]
  - Several learning results
  - No software available
- Relational Bayesian networks (RBNs) [Jaeger 2001]
  - Software available: Primula
- Bayesian logic programs (BLPs) [Kersting & De Raedt 2001]
- Multi-entity Bayes nets (MEBN) [Laskey & da Costa 2005]



## Stochastic Programming Languages

- Idea: Let random coin flips drive a deterministic program, get distribution over outputs
- IBAL: Stochastic functional (Lisp-like) language [Pfeffer 2001]
- Stochastic Prolog:
  - Probabilistic Horn abduction [Poole 1993]
  - PRISM [Sato & Kameya 1997]



## RPMs Based on Undirected Graphical Models

- Relational Markov networks [Taskar *et al.* 2002]
- Markov logic networks (MLNs) [Richardson & Domingos 2006]
- Benefits compared to directed models
  - Don't have to worry about acyclicity
  - Specify weights on features, not full CPDs
- Drawbacks
  - Feature weights harder to interpret than CPDs
  - Parameters must be estimated jointly

## Summary

- Modeling unknown objects is essential
- BLOG models define probability distributions over possible worlds with
  - Varying sets of objects
  - Varying mappings from observations to objects
- MCMC can provide effective inference for models of this kind

43

## References


- Milch, B., Marthi, B., Russell, S., Sontag, D., Ong, D. L., and Kolobov, A. (2005) "BLOG: Probabilistic Models with Unknown Objects". In *Proc. 19th Int'l Joint Conf. on AI*, pages 1352-1359.
- Milch, B., Marthi, B., Sontag, D., Russell, S., Ong, D. L., and Kolobov, A. (2005) "Approximate inference for infinite contingent Bayesian networks". In *Proc. 10th Int'l Workshop on AI and Statistics*.
- Milch, B. and Russell, S. (2006) "General-purpose MCMC inference over relational structures". In *Proc. 22nd Conf. on Uncertainty in AI*, pages 349-358.
- Pasula, H., Marthi, B., Milch, B., Russell, S., and Shpitser, I. (2003) "Identity uncertainty and citation matching". In *Advances in Neural Information Processing Systems 15*, MIT Press, pages 1401-1408.
- Lawrence, S., Giles, C. L., and Bollacker, K. D. (1999) "Autonomous citation matching". In *Proc. 3rd Int'l Conf. on Autonomous Agents*, pages 392-393.
- Wellner, B., McCallum, A., Feng, P., and Hay, M. (2004) "An integrated, conditional model of information extraction and coreference with application to citation matching". In *Proc. 20th Conf. on Uncertainty in AI*, pages 593-601.


44

## References


- Pasula, H., Russell, S. J., Ostland, M., and Ritov, Y. (1999) "Tracking many objects with many sensors". In *Proc. 16th Int'l Joint Conf. on AI*, pages 1160-1171.
- Oh, S., Russell, S. and Sastry, S. (2004) "Markov chain Monte Carlo data association for general multi-target tracking problems". In *Proc. 43rd IEEE Conf. on Decision and Control*, pages 734-742.
- Koller, D. and Pfeffer, A. (1998) "Probabilistic frame-based systems". In *Proc. 15th AAAI National Conf. on AI*, pages 580-587.
- Friedman, N., Getoor, L., Koller, D.,and Pfeffer, A. (1999) "Learning probabilistic relational models". In *Proc. 16th Int'l Joint Conf. on AI*, pages 1300-1307.
- Gilks, W. R., Thomas, A. and Spiegelhalter, D. J. (1994) "A language and program for complex Bayesian modelling". *The Statistician* 43(1):169-177.
- Jaeger, M. (2001) "Complex probabilistic modeling with recursive relational Bayesian networks". *Annals of Mathematics and AI* 32:179-220.
- Kersting, K. and De Raedt, L. (2001) "Adaptive Bayesian logic programs". In *Proc. 11th Int'l Conf. on Inductive Logic Programming*.


45

## References


- Laskey, K. B. and da Costa, P. C. G. (2005) "Of Starships and Klingons: First-order Bayesian logic for the 23rd century" In *Proc. 21st Conf. on Uncertainty in AI*, pages 346-353.
- Mjolsness, E. and Yosiphon, G. (to appear) "Stochastic process semantics for dynamical grammars". *Annals of Mathematics and AI*.
- Pfeffer, A. (2001) "IBAL: A probabilistic rational programming language". In *Proc. 17th Int'l Joint Conf. on AI*, pages 733-740.
- Poole, D. (1993) "Probabilistic Horn abduction and Bayesian networks". *Artificial Intelligence* 64(1):81-129.
- Sato, T. and Kameya, Y. (1997) "PRISM: A symbol-statistical modeling language". In *Proc. 15th Int'l Joint Conf. on AI*, pages 1330-1335.
- Taskar, B., Abbeel, P., and Koller, D. (2002) "Discriminative probabilistic models for relational data". In *Proc. 18th Conf. on Uncertainty in AI*, pages 485-492.
- Richardson, M. and Domingos, P. (2006) "Markov logic networks". *Machine Learning* 62:107-136.


46