COMPSCI 276: Projects for Belief Networks Fall, 2007

Rina Dechter

November 13, 2007

Students can do individual projects or can team in groups of 2 for a project. You will need to provide 20-30 minute prsentation of your project in class and to write a report which is due i the finals week. Presentations will take place in the 10th and the 11th week.

First order probabilistic languages

1. **The Markov Logic: a first order probabilistic language.** Study the language, experiment with the software, model 1-2 problems of your own. Below are some pointers (see also the project wb page).

   - P. Domingos et al. Markov Logic. In L. De Raedt, P. Frasconi, K. Kersting and S. Muggleton (eds.), Probabilistic Inductive Logic Programming. New York: Springer. To appear.
     http://www.cs.washington.edu/homes/pedrod/papers/pilp.pdf
   - A shorter and less up-to-date version is "P. Domingos et al. "Unifying Logical and Statistical AI" Proceedings of the Twenty-First National Conference on Artificial Intelligence (pp. 2-7), 2006. Boston, MA: AAAI Press."
     http://www.cs.washington.edu/homes/pedrod/papers/aaai06c.pdf
   - Intro slides (AAAI-06 invited talk):
     http://www.cs.washington.edu/homes/pedrod/ulsai.ppt
   - Longer tutorial slides (AAAI-07 tutorial):
     http://www.cs.washington.edu/homes/pedrod/psrai.ppt
   - There's a video tutorial, text tutorial with examples, software, MLNs, datasets, etc., on the Alchemy Web site: http://alchemy.cs.washington.edu

2. **BLOG (Bayesian Logic: a first order probabilistic language.** Study the language, experiment with the software, model problems of your own. Below are some pointers (see also the project's webpage)

   - The software and papers are available for use at
     http://people.csail.mit.edu/milch/blog/index.html

- Slides http://www.cs.berkeley.edu/ russell/talks/russell-irvine06.ppt
- The IJCAI 2005 paper.

Exact and approximate Reasoning Algorithms: Study, evaluate and extend

You can develop your own code, or more economically use existing code/packages at UC-Irvine (we have http://graphmod.ics.uci.edu/group/Software and REES software) or anything you can find online. For each projet you will need to read papers and make yourself familiar with the relevant algorithms. You will also experiment with various benchmarks. The projects will be guided by myself and graduate students Radu Marinescu and Vibhav Gogate. Some of the projects can evolve into a larger scale research that can lead to a conference paper.

1. AND/OR decision diagrams for Bayesian networks. YOu should read and present recent paper on this topic: "Robert Mateescu and Rina Dechter. "AND/OR Multi-Valued Decision Diagrams (AOMDDs) for Weighted Graphical Models" http://www.ics.uci.edu/In UAI'07.

   and run additional experiments using code that will be provided. You can also extend existing code.

2. **Comparing exact and local search for constraint optimization.** Guided by Radu.

   Take Radu's AOBB code which is a branch and bound algorithm for the mpe task as well as his local search algorithm for mpe whose codes are available and evaluate and compare on the Bayesian networks that we provide.

3. **Algorithm evaluation for P(e) on various benchmarks.** Guided by Vibhav. Evaluate sampling vs exact algorithms for P(e) on various benchmarks. Focus on linkage analysis.

4. **Develop bounding algorithm for P(e) using the mini-bucket scheme.** Guided by vibhav Gogate or Radu Marinescu.

5. **Use multiple mini-bucket heuristics for mpe.** Guided by Radu. Extend the branch and bound algorithm AOBB which is mini-bucket heuristics to work with multiple heuristics created by randomizing the mini-bucket scheme. You will work with existing code.

6. **Efficient combine-eliminate algorithm for relational probabilistic functions.**

   Many algorithms in graphical models that use variable elimination or join-tree/junction-tree clustering involve the atomic operator of "combine-eliminate". In Bayesian inference the two functions are combined by product and variables are

eliminated by summation or maximization/minimization. An efficient implementation of this combine-eliminate is of utmost importance. The project is to develop algorithm(s) and code for doing the combine-eliminate (produc-sum or product-max/min) when the input functions are specified as either: 1. relations, 2. decision trees, 3. Ordered decision diagrams. See, "Bayesian Inference in the Presence of Determinism", Larkin and Dechter In AI and Statistics, AI-STAT, 2003. http://www.ics.uci.edu/ dechter/publications/ Another relevant paper is Kalev Kask, Rina Dechter, Javier Larrosa and Avi Dechter. "Unifying Cluster-Tree Decompositions for Reasoning in Graphical Models". In Artificial Intelligence Journal, 2005. http://www.ics.uci.edu/

7. **Bounding the time and space complexity of a algorithms on a given problem instance for finding mpe or P(e)**. This can be done based on tree and hypertree decomposition (talk to me).

8. Look at the paper: "Edge Deletion, Edge Correction, and the Bethe Free Energy" by Arthur Choi and Adnan Darwiche
http://reasoning.cs.ucla.edu/fetch.php?id=68&type=pdf . Compare with belief propagation algorithms on REES or with your own code of belief propagation.

9. **Bayesian network Preprocessor.** Guided by Vibhav Gogate.

   Input: A Bayesian network with some zero probabilities and an evidence set. Output: A network in which (a) irrelevant subnetworks are removed (b) Domains are pruned by enforcing some level of constraint propagation. (c) Some other sound pre-processing that you could think of. For ideas read relevant paper by David Allan. Use Ergo network format described on our software website.

10. Propose a theoretical or empirical question you want to investigate.