Overview

In this seminar, we will examine several families of approximate inference algorithms, and consider how to interpret them in the context of cognitive modelling.

Computational cognitive modelling is often pursued in the framework of Marr’s levels of analysis. In this framework, cognitive systems are information-processing systems, and can be analyzed at three levels of abstraction:

1. The computational level: A precise description of the goal of the system.
2. The algorithmic/representational level: A precise description of the representations, and operations on those representations, that the system uses to achieve its goal.
3. The implementation level: A precise description of how those representations and operations are physically realized, such as assemblies of biological neurons.

This framework relies on the observation that the same function (a computational level description) may be computed by many different algorithms, and the same algorithm may be implemented using many different physical substrates (gears, vacuum tubes, silicon transistors, biological neurons, etc.). This approach is methodologically appealing because each level has different constraints. For example, the computational level is primarily constrained by the nature of the task, but abstracts away from the biochemical constraints that are important at the implementation level.

Cognitive modelling work typically focuses on the computational level, and uses whatever algorithm is most practical for evaluating the computational level hypothesis. For natural language learning, however, exact inference is typically intractable and approximations must be used. With an approximate algorithm, we are no longer evaluating the computational level proposal exactly: properties of the approximation may ‘leak’ into the behavior of our system. Moreover, recent work has developed increasingly approximate algorithms to enable inference in more elaborate models on larger datasets. In this seminar, we will consider several families of approximate inference algorithms with the intention of understanding how different approximations may affect system performance, beyond the properties of the computational-level proposal.

Course Set-up

Class discussions

Class meetings will mostly consist of paper discussions. You will be asked to lead the discussion for six meetings throughout the semester. To prepare for leading a discussion, you should read the paper in advance, identify the main points, and record points of confusion – you do not need to be an ‘expert’ on the paper. If you like, you may prepare slides, but this is not required.

When you are not leading a discussion, you should read the paper, and either post a question you have about the paper on the UBLearns message board, or respond to somebody else’s question.

Final paper

There are two options for the final paper:

- Programming option: Take one natural language model, implement three approximate inference algorithms for it, and assess how the different approximations affect system behavior. What can you learn about this model and the underlying phenomenon?
• Non-programming option: Review and integrate papers that have modelled some natural language phenomenon using several different approximation algorithms. What have we learned about the phenomenon across the different approximations?

**Grading breakdown**

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message board participation</td>
<td>16%</td>
</tr>
<tr>
<td>Leading discussions</td>
<td>24%</td>
</tr>
<tr>
<td>Final paper</td>
<td>60%</td>
</tr>
</tbody>
</table>

**Week 1:** Set-up and exact inference

Aug 29: Course intro and paper assignments
Aug 31: Exact inference *(Marr, 1982, chapter on UBLearns)*
Sep 02: Exact inference in graphical models *(Kschischang et al., 2001)*

**Week 2:** Heuristic inference

Sep 05: Labor day – no meeting
Sep 09: Minimum description length for morphology induction *(Goldsmith, 2001)* – Dan Fox

**Week 3:** Variational inference

Sep 12: Another minimum description length for morphology induction *(Creutz and Lagus, 2002)* – Yu Li
Sep 14: Expectation-maximization and variants *(Neal and Hinton, 1999)* – Dan Fox
Sep 16: The Dependency Model with Valence *(Klein and Manning, 2004)* – Ali Alshehri

**Week 4:** Scalable variational inference

Sep 19: Latent Dirichlet Allocation *(Blei et al., 2003)* – Hao Sun
Sep 21: VB for PCFGs *(Kurihara and Sato, 2006), (Jones, 2013)* – Dan Fox
Sep 23: VB for PCFGs (cont’d) – John Pate

**Week 5:** Scalable variational inference (Cont’d)

Sep 26: Stochastic VB (for LDA) *(Hoffman et al., 2013)* – Yu Li
Sep 28: Collapsed Variational Bayes (for LDA) *(Teh et al., 2007)* – Erika Bellingham
Sep 30: CVB for PCFGs *(Wang and Blunsom, 2013)* – Hao Sun

**Week 6:** Sampling intro

Oct 03: Streaming VB (for LDA) *(Broderick et al., 2013)* – Dan Fox
Oct 05: Stochastic VB for Combinatory Categorial Grammar *(Kwiatkowski et al., 2012)* – Yu Li
Oct 07: Gibbs sampling with tears *(Knight, 2009)* – John Pate

**Week 7:** Sampling applications

Oct 10: Gibbs sampling with tears *(Knight, 2009)* – John Pate
Oct 12: Gibbs sampling with tears *(Knight, 2009)* – John Pate
Oct 14: Sampling for HMMs *(Goldwater and Griffiths, 2007)* – Hao Sun

**Week 8:** Sampling applications
Oct 17: Sampling for PCFGs (Johnson et al., 2007) – Ali Alshehri
Oct 19: Sampling for word segmentation (Goldwater et al., 2009) – Yu Li
Oct 21: Sampling for Tree Substitution Grammar (Blunsom and Cohn, 2010) – Erika Bellingham

Week 9: More sampling, and a couple of particle filter papers

Oct 24: Particle filters for word segmentation (Börschinger and Johnson, 2011), (Börschinger and Johnson, 2012) – Ali Alshehri
Oct 26: Word segmentation and vowel category acquisition jointly (Feldman et al., 2013) – TBD
Oct 28: Particle filters to model garden paths (Levy et al., 2009) – Yu Li

Week 10: Maximum Entropy models and Neural Nets

Oct 31: Win-Stay Lose-Sample (Bonawitz et al., 2014) – Erika Bellingham
Nov 02: Maximum Entropy intro (Berger et al., 1996) – John Pate
Nov 04: Using Maximum Entropy models in generative models (Berg-Kirkpatrick et al., 2010) – Dan Fox

Week 11: Theano tutorial

Nov 07: Neural networks, (Mackay, 2003, ch. 39) – John Pate
Nov 09: – John Pate
Nov 11: – John Pate

Week 12: Neural language models

Nov 14: Theano tutorial (wrap-up) – John Pate
Nov 16: Maximum Entropy for phonology (Hayes and Wilson, 2008) – Dan Fox
Nov 18: Word segmentation with Recurrent Neural Nets (Christiansen et al., 1998) – Ali Alshehri

Week 13: Neural language models

Nov 21: Neural language models (Mnih and Hinton, 2007) – Erika Bellingham
Nov 23: Fall recess
Nov 25: Fall recess

Week 14: Learning rate optimization

Nov 28: Negative Sampling (Mikolov et al., 2013) – Hao Sun
also see (Goldberg and Levy, 2014) for more info (very short).
Nov 30: Noise Contrastive Estimation (Mnih and Teh, 2012) – Ali Alshehri
See (Dyer, 2014) for more info (very short).
Dec 02: Adam (Kingma and Ba, 2015) – Yu Li

Week 15: Overflow dates

Dec 05: Adagrad (Duchi et al., 2011) – Erika Bellingham
also see http://sebastianruder.com/optimizing-gradient-descent/index.html#adagrad
Dec 07: Adadelta (Zeiler, 2012) – Hao Sun
also see http://sebastianruder.com/optimizing-gradient-descent/index.html#adadelta
Dec 09: Overflow dates
References


Jones, B. (2013). Mean field variational bayes for context free formalisms.


