

VICTORIA UNIVERSITY OF WELLINGTON  
*Te Whare Wānanga o te Ūpoko o te Ika a Māui*



School of Engineering and Computer Science  
*Te Kura Mātai Pūkaha, Pūrorohiko*

PO Box 600  
Wellington  
New Zealand

Tel: +64 4 463 5341  
Fax: +64 4 463 5045  
Internet: [office@ecs.vuw.ac.nz](mailto:office@ecs.vuw.ac.nz)

## **Richer Restricted Boltzmann Machines**

Max Godfrey

Supervisors: Marcus Freen

Submitted in partial fulfilment of the requirements for  
Bachelor of Engineering with Honours.

### **Abstract**

TODO WRITE THE ABSTRACT



# Acknowledgments

Any acknowledgments should go in here, between the title page and the table of contents. The acknowledgments do not form a proper chapter, and so don't get a number or appear in the table of contents.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Problem	1
1.1.1	Deep Belief Networks can achieve state of the art performance	1
1.1.2	DBNs have no mechanism for separating sources	1
1.1.3	Restricted Boltzmann Machines cannot separate sources either	1
1.1.4	Sigmoid Belief Networks; Intractably rich in practice	1
1.2	Solution	2
1.2.1	Trading tractability for Source Separation	2
<b>2</b>	<b>Backgroud</b>	<b>3</b>
2.1	Source Separation	3
2.1.1	An example, the cocktail party problem	3
2.2	Generative Models	3
2.2.1	Modeling building blocks, observable and hidden variables	3
2.2.2	PGMs as a tool reasoning about generative models	4
2.3	Sampling and inverting the model	4
2.3.1	Gibbs sampling, a subset of Markov Chain Monte Carlo	4
2.3.2	Reconstructions, visualising what the model has learnt	5



# Figures

1.1	A graphical representation showing the proposed generative model for capturing two causes, the ORBM. . . . .	2
2.1	An example PGM, showing an observed variable 'A' and it's hidden cause 'B'. . . . .	4





# Chapter 1

## Introduction

### 1.1 Problem

#### 1.1.1 Deep Belief Networks can achieve state of the art performance

Deep Belief networks are powerful models that have proven to achieve state of the art performance in many domains. For instance a non-exhaustive list is image classification, dimensionality reduction, natural language recognition, Document classification, Semantic Analysis and .

DBNs capture non-linear interactions between low level features, in the context of image classification the lower layers can capture image filters.

#### 1.1.2 DBNs have no mechanism for separating sources

Despite a DBNs expressiveness, there is no way to extract these interactions. If an input has multiple sources then the complex combination is instead learnt, the network has no mechanism for extracting multiple causes. This is the motivation for this project, to be able to separate the sources of data in a new model.

[1]

#### 1.1.3 Restricted Boltzmann Machines cannot separate sources either

Restricted Boltzmann Machines are two layer, fully connected, unsupervised neural networks. DBNs are constructed by stacking RBMs. Being the building block of the powerful DBN, RBMs are a natural starting point for representing multiple sources. RBMs make the assumption that the features of the input data are dependant in the prior, as they are independant in the posterior. The latter makes them tractable to use in practice, but also means they model/encode a single representation. Again using the example of images, an input image will map to a single representation, again there is a lack of mechanism for modelling sources that are acting independantly.

#### 1.1.4 Sigmoid Belief Networks; Intractably rich in practice

The Sigmoid Belief Network, the parameterized version of a Bayesian/Belief network appears as a natural choice for modelling independant sources in that it makes a polar assumption to the RBM; – Warning Semicolon Use – Every feature has an independant cause. The sigmoid belief networks assumption could capture data that has multiple sources, but this is intractable in practice.

## 1.2 Solution

### 1.2.1 Trading tractability for Source Separation

Frean and Marsland propose a generative model that aims to trade a small amount of the RBMs performance for richness, finding a middle ground between the sigmoid belief network and the restricted boltzmann machine. Frean and Marsland also propose an algorithm to invert this model, separating the sources of an input.

The new generative model, referred to onwards as an ORBM, uses an RBM to model each source and a sigmoid belief network to capture their combination to form data. This project explores the ORBM use for separating two causes.

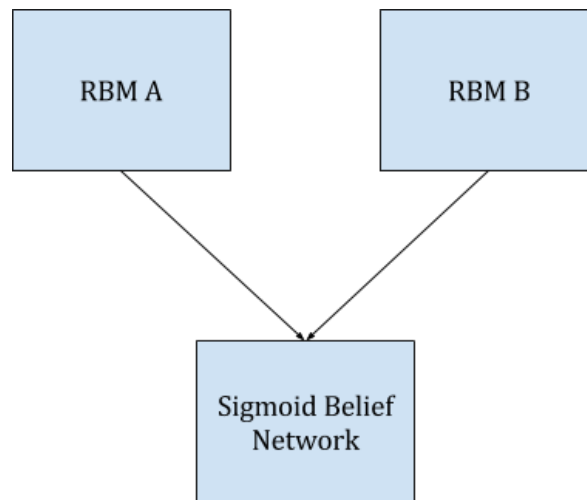


Figure 1.1: A graphical representation showing the proposed generative model for capturing two causes, the ORBM.

Given the proposed model and algorithm, this project answers the following questions:

- Can this model encode data comprised of more than one cause as it's constituted causes? That is, can the model and new algorithm for inverting it, perform source separation.
- Is the ORBMs two cause structure to rich to be tractible in practice?

# Chapter 2

## Backgroud

As the ORBM builds on the previous work of Restricted Boltzmann Machines and Sigmoid Belief networks, the concepts and previous work in source separation, generative models as well as background on RBMs and Sigmoid Belief Networks need to be introduced.

### 2.1 Source Separation

#### 2.1.1 An example, the cocktail party problem

A famous example that illustrates the idea of source separation is the cocktail party problem. At a cocktail party many conversations are taking part at the same time, creating noise, a composition of all the conversations. Despite this, a partygoer is able to separate their conversation from cacophany, separating the sources.

The applications of source separation are far wider than talkative partygoers. In the field of signal processing

TODO Draft Points Still to write TODO

### 2.2 Generative Models

Generative models are a powerful way to model data.

TODO Draft Points Still to write TODO

(TODO-GRAB-THOSE-GENERATIVE-MODEL-USES-CITATIONS) Basically justify generative models.

The ORBM proposed in this project aims to represent data generated by two indepedanlty acting causes and does so by combining two existing generative models, the Restricted Boltzmann Machine, and the Sigmoid Belief Network.

#### 2.2.1 Modeling building blocks, observable and hidden variables

Generative models are comprised of variables, often referred to as units. Some of these variables are observed, that is their state is known. These are often referred to as the 'visible' units and are used to represent the training data. For example in the image domain, the visible units correspond to the pixels of the image.

The variables that are not observed, are latent variables, often referred to as 'hidden units' as they are not observed.

Connections between units are used to encode relationships between the variables, where the relationship may be causal, such as in a Sigmoid Belief network or an encoding / representation in the Restricted Boltzmann Machine.

### 2.2.2 PGMs as a tool reasoning about generative models

**TODO Draft Points Still to write TODO**

Probabilistic Graphical Models or PGMs for short, are an expressive way to represent a collection of related, stochastic variables. If the graph is directed then the edges represent causation, this is also referred to a Bayesian network. Conversely, if the graph was undirected then edges represent a dependancy or mapping. Throughout this report, RBMs, Sigmoid Belief Networks and the proposed ORBM will be shown in this format.

Figure 2.1 shows an abstract example of a directed PGM, where B is the underlying cause of A, we cannot observe B directly, instead it's state is represented as a 'belief' or a probability of being in a given state.

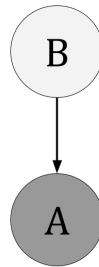


Figure 2.1: An example PGM, showing an observed variable 'A' and it's hidden cause 'B'.

## 2.3 Sampling and inverting the model

Sampling is the process of drawing samples from a distribution. It is used when the distrubution we want samples from is intractable to calculate analytically. Sampling is required to train generative models, as often the gradient to be climbed / descended involves calculating a probability in the generative model.

**TODO Draft Points Still to write TODO**

- Inference is the process of given reasoning about what we do not know given that of which we do know.
- In a Generative Model this amounts to the Posterior

### 2.3.1 Gibbs sampling, a subset of Markov Chain Monte Carlo

- The importance of Markov Chains and mixing time are crutial in this project

Gibbs sampling is a special case of Markov Chain Monte Carlo, a technique for drawing sampling from a complex distribution. The probability mass (the joint distribution) of a generative model is a common use case for Gibbs sampling, and is used for performing inference in the RBMs, Sigmoid Belief Networks and in the ORBM. Being a specific case of Markov Chain Monte Carlo (refferred to as MCMC).

## Mixing Time

MCMC methods require a ‘mixing’ phase to ensure convergence, that is that the sample is being drawn from a representative part of the desired distribution. This is part of the trade off the ORBM attempts to make, as a mixing time is introduced that is not present in the RBM.

### 2.3.2 Reconstructions, visualising what the model has learnt

Generative Models can create an internal representation given an input. They can also generate a faux input given an internal representation. Performing one Gibbs iteration, that is sampling from the hidden units given an input  $P(\tilde{h}|\tilde{v})$  and then taking the generated hidden state and generating a faux input. The model tries to reconstruct the input.

#### Fanstasies of the model

##### TODO Draft Points Still to write TODO

In the same way that a generative model uses reconstructions to try and recreate the supplied input based purely on how it’s represented that input, performing many, many (greater than 100) Gibbs iterations with no input pattern clamped allows the reconstructions to explore the probability mass that the model has built up during training. Sampling from these wanderings creates what are referred to as ‘fantasies’ or ‘dreams’. These give a sense of what the model has learnt, and can act a smoke test for if the model has actually capture anything. (TODO-CITE-PAPER-WITH-MNIST-DREAM-EVALUATION, they were crappy).



# Bibliography

[1] History of SciPy. [http://wiki.scipy.org/History\\_of\\_SciPy/](http://wiki.scipy.org/History_of_SciPy/). Accessed: 2015-07-20.