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

Logical Neural Networks: Opening the black box

Daniel Thomas Braithwaite

Supervisor: Marcus Frean

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

Abstract

Neural networks have been shown to be universal function approximators and have been employeed to solve a number of non-trivial problems. The downside to these networks is that they are very diffcult (if not impossible) to understand. We aim to show that building NN's out of Noisy logic gates are also universal function approximators, have comparable peformance to regular NN's and the added benifit of being understandable to humans

1. Introduction

Neural Networks (NN's) peform exceptionally well over a wide range of different problems. However once trained these networks become a black box, near impossible for a human to understand what features the network is using to solve the problem presented to it. Logical Neural Networks (LLN's), that is NN's with logic based activation functions, have been shown to provide a more understandable representation. However LLN's have been neglected and not a lot of effort has been put into developing them.

2. The Problem

Research done previously into LLN's has shown that using a network with layers consisting of Noisy-OR and Noisy-And neurons can achieve a reasonable accuracy on the MINST dataset while also providing a more understandable network. To begin this project we take a step back to ensure we have the right approach. A network built out of neurons that resembel logic gates we would expect should be able to learn any given boolean formula, failure to do so would indicate a fundimental problem with our approach.

Ideally we would give a mathematical proof showing that LLN's are (or are not) universal boolean function approximators. If LLN's are this would provide confidence for them also being universal function approximators in the general case. If we are not able to prove (or provide evedence via experiments) that LLN's are universal function approximators then there uses would be very limited.

3. Proposed Solution

Before we can start we must address a fundimental issue with our existing setup. Namely our choice of "gates" is not a functionally complete set. Trying to learn a boolean function with only AND and OR might work but it will severly limit us. We proposes to implement a number of different sets of functionally complete neurons, such as NAND, NOR. Ideally we would find that they all give the same peformance but given we are modeling our descrete gates with a continous function we could see unexpected results.

After experementing with LLN's for approximating boolean functions we would have a good idea for whether they are universal boolean function approximators or not, intuatively it would make sense for them to be, given that we would be using a functionally complete set of logic gates (which we know in their descrete form can represent any boolean expression). However as disussed before, given we are approximating our descrete gates as continous ones, the same principles may not hold. We would first need to investigate what effects our continous model has on the propertys of logic gates. Ideally we would design these gates in a way that maintains these nice propertys, e.g. we would want NOT(AND) = NAND.

4. Evaluating your Solution

5. Resource Requirements