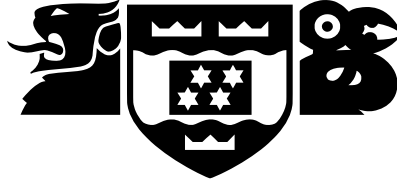


VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wānanga o te Ūpoko o te Ika a Māui



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

some kind of rnn/tensor mess

Paul Francis Cunninghame Mathews

Supervisors: Marcus Frean and David Balduzzi

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

Abstract

A short description of the project goes here.

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

1	Introduction	1
2	Background and Related Work	3
2.1	Background	3
2.1.1	Neural Networks	3
2.2	Related Work	3
2.2.1	Long Time Dependencies	3
2.2.2	Memory	3
2.2.3	Tensors in Neural Networks	3
3	Tensors	5
3.1	Definitions	5
3.2	Bilinear Products	5
3.3	Tensor Decompositions	5
3.3.1	CANDECOMP/PARAFAC	5
3.3.2	Tensor Train, Tucker	5
3.4	Learning decompositions by gradient descent	5
4	Proposed Architectures	7
4.1	Incorporating tensors for expressivity	7
4.2	Gates and Long Time Dependencies	7
4.3	Proposed RNNs	7
5	RNN Experiments (better title plz)	9
5.1	Synthetic Tasks	9
5.1.1	Addition	9
5.1.2	Variable Binding	9
5.1.3	MNIST	9
5.2	Real-world Data	9
5.2.1	Polyphonic Music	9
5.2.2	PTB	9
5.2.3	War and Peace	9
6	Conclusions	11

Figures

Chapter 1

Introduction

This chapter gives an introduction to the project report.

In Chapter ?? we explain how to use this document, and the `vuwproject` style. In Chapter ?? we say some things about \LaTeX , and in Chapter 6 we give our conclusions.

Chapter 2

Background and Related Work

2.1 Background

2.1.1 Feed-Forward Neural Networks

This is to be pretty brief. Cover useful things – what they look like, gradient descent (in brief) and outline some results on the expressive power.

2.1.2 Recurrent Neural Networks

What they look like, what they can do, how to train them (and when/why it doesn't work). LSTMs and GRUs? Or in related work?

2.2 Related Work

2.2.1 Long Time Dependencies

Summary of approaches to fix vanishing gradients.

2.2.2 Memory

Summary of approaches to augmenting RNNs with extra memory, or other approaches to better use memory.

2.2.3 Tensors in Neural Networks

Including gated networks, MRNN and so on.

Chapter 3

Tensors

This chapter discusses some necessary /useful multi-linear algebra which we use later.

3.1 Definitions

3.2 Bilinear Products

3.3 Tensor Decompositions

3.3.1 CANDECOMP/PARAFAC

3.3.2 Tensor Train, Tucker

3.4 Learning decompositions by gradient descent

Multiplicative dynamics = instability?

Chapter 4

Proposed Architectures

4.1 Incorporating tensors for expressivity

4.2 Gates and Long Time Dependencies

4.3 Proposed RNNs

Chapter 5

RNN Experiments (better title plz)

5.1 Synthetic Tasks

Pathological, exercise specific features of the architecture.

5.1.1 Addition

5.1.2 Variable Binding

5.1.3 MNIST

is really dumb

5.2 Real-world Data

Mostly testing rank as regulariser

5.2.1 Polyphonic Music

5.2.2 PTB

5.2.3 War and Peace

Chapter 6

Conclusions

The conclusions are presented in this Chapter.

