



NTNU – Trondheim
Norwegian University of
Science and Technology

Recurrent Neural Network for Recommendation and User Representation

Ole Steinar Lillestøl Skrede

November 2016

Department of Computer and Information Science
Norwegian University of Science and Technology

Supervisor: Massimiliano Ruocco

Abstract

Contents

1	Introduction	1
2	Motivations	2
3	State of the Art	3
4	Challanges	4
5	Ideas	5
6	Experimentation	6
7	Analysis of Results	7
8	Conclusion	8
9	Further Work	9
A	List of Acronyms	10
B	appendix-b	11
	Bibliography	12

Chapter 1

Introduction

BLa bla bla

Chapter 2

Motivations

Chapter 3

State of the Art

A lot of this should be in the introduction instead
Much of this information is sourced from [1] and should maybe be referenced.
However (what I have used so far) is pretty obvious (?) stuff for those who know
a bit about RNNs, so might not be needed.

Neural networks achieve great results in many different machine learning tasks these days. One example is the use of deep neural networks in Google DeepMind's AlphaGo program, which became the first Computer Go program that to beat a professional human [2].

The growth in computing power has made neural networks more applicable. Also, they work well in some cases where other methods struggle.

Standard neural networks are not well suited to handle cases where the training and test data are not independent. And they are not suited for inputs and outputs of varying length either. Recurrent neural networks works better in these cases. If neural networks already have state of the art performance on a problem then it might be worth looking into using recurrent neural networks to further increase the performance. Standard neural networks are not the only model that lacks the ability to evaluate data points in sequence, for example support vector machines and logistic regression do not have a sense of time either.

Markov models can represent time dependencies, but becomes infeasible to use when the number of dependencies grow big. RNNs have the advantage that they can represent an exponential amount of hidden states in the hidden nodes.

For more detail on the state of the art on RNNs, we refer to Zachary Chase Lipton's comprehensive paper "A Critical Review of Recurrent Neural Networks for Sequence Learning" [1]

Chapter 4

Challanges

sdfgdf

Chapter 5

Ideas

- Encode more information in input to RNN? E.g. time.
- Use RNN to learn embedding space: SESSION_2_VEC
- Combine session output vectors to a user vector: USER_2_VEC
- Use word2vec to represent title of book. Give this as input to RNN (Or use vector representation of other types of input)
- Use LSTM (long term short term memory) or BRNN instead of GRN

Chapter 6

Experimentation

sdfg

Chapter 7

Analysis of Results

dsfd

Chapter 8

Conclusion

sdfgsdfg

Chapter 9

Further Work

sdfgsdfg

Appendix A

List of Acronyms

RNN Recurrent Neural Network

Appendix B

appendix-b

Bibliography

- [1] Zachary Chase Lipton. A critical review of recurrent neural networks for sequence learning. *CoRR*, abs/1506.00019, 2015.
- [2] BBC. Google achieves ai 'breakthrough' by beating go champion. <http://www.bbc.com/news/technology-35420579>.