THESIS TITLE

MSc Thesis *(Afstudeerscriptie)*

written by

Krsto Proroković
(born March 12, 1993 in Kotor, Montenegro)

under the supervision of **Dr Germán Kruszewski** and **Dr Elia Bruni**, and submitted to the Board of Examiners in partial fulfillment of the requirements for the degree of

MSc in Logic

at the *Universiteit van Amsterdam.*

Date of the public defense:	**Members of the Thesis Committee:**
Date of the defense goes here	Committee goes here

INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

Abstract

Abstract goes here

Acknowledgents

Contents

Chapter 1

Introduction

Chapter 2

Background

2.1 Recurrent Neural Networks

Recurrent neural networks [reference] are ... for processing sequences ... They ... handwriting recognition, speech ... In this chapter we ... For a more detailed treatment we point the reader to [reference to Alex Graves' book]

2.1.1 Vanilla Recurrent Neural Networks

We start with a simple vanilla RNN model. ... given with:

- Input to hidden ... $W_x \in \mathbb{R}^{d_h \times d_x}$
- Hidden to hidden ... $W_h \in \mathbb{R}^{d_h \times d_h}$
- Bias term ... $b_h \in \mathbb{R}^{d_h}$
- Activation function $\phi : \mathbb{R} \to \mathbb{R}$
- Hidden to output ... $W_y \in \mathbb{R}^{d_y \times d_y}$
- Bias term ... $b_y \in \mathbb{R}^{d_y}$
- Initial hidden state $h^{(0)} \in \mathbb{R}^{d_h}$ (usually set to ...)

...

$$
\begin{aligned}
h^{(t)} &= \phi\left(W_h h^{(t-1)} + W_x x^{(t)} + b_h\right) \\
y^{(t)} &= \mathrm{softmax}\left(W_y h^{(t)} + b_y\right)
\end{aligned}
$$

... Here we will be mostly interested in sequence classification, i.e. we will only care about $y^{(T)}$ ...

Example Suppose that the input sequence consists of ... Consider a vanilla RNN given with

$$
W_x = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad W_h = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad b = \begin{bmatrix} 0 \\ 0 \end{bmatrix}
$$

While the previous example ... rather simple function ... RNNs are universal [reference Siegelmann and Sontag]

2.1.2 Backpropagation Through Time and Vanishing Gradient

...

2.1.3 Long-Short Term Memory Networks

One way to deal with vanishing gradient is to use gated RNNs.
One way to deal with vanishing gradient are Long-Short Term Memory Networks [2] ...

$$\begin{aligned}
\begin{bmatrix} f_t \\ i_t \\ o_t \\ g_t \end{bmatrix} &= W_x x_t + W_h h_t + b \\
c_t &= \sigma(f_t) \odot c_{t-1} + \sigma(i_t) \odot \tanh(g_t) \\
h_t &= \sigma(o_t) \odot \tanh(c_t)
\end{aligned}$$

LSTM has been successful in many applications, such as ...

2.1.4 Gated Recurrent Units

Another gated recurrent architecture are Gated Recurrent Units (GRU) ... [1]

$$\begin{aligned}
r_t &= \sigma(W_{xr} x_t + W_{hr} h_{t-1} + b_r) \\
z_t &= \sigma(W_{xr} x_t + W_{hz} h_{t-1} + b_z) \\
n_t &= \tanh(W_{xn} x_t + r_t(W_{hn} h_{t-1} + b_n)) \\
h_t &= (1 - z_t) n_t + z_t h_{t-1}
\end{aligned}$$

2.1.5 Embeddings

2.1.6 Attention Models

Chapter 3

Experiments (title subject to change)

Bibliography

[1] Kyunghyun Cho et al. “Learning phrase representations using RNN encoder-decoder for statistical machine translation”. In: *arXiv preprint arXiv:1406.1078* (2014).

[2] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory”. In: *Neural Computation* 9.8 (1997), pp. 1735–1780.