KU LEUVEN

FACULTEIT INGENIEURSWETENSCHAPPEN

# Building a state of the art speech recogniser

Moritz Wolter

Thesis submitted for the degree of Master of Science in Mathematical Engineering

**Thesis supervisor:**
Prof. dr. ir. Patrick Wambacq

**Assessor:**
Prof. dr. ir. Johan Suykens

**Mentor:**
Ir. Vincent Renkens

Academic year 2015 – 2016

# Preface

I would like to thank everybody who kept me busy the last year, especially my promotor and my assistants. I would also like to thank the jury for reading the text. My sincere gratitude also goes to my family for supporting me trough my studies.

*Moritz Wolter*

# Contents

# Abstract

The `abstract` environment contains a more extensive overview of the work. But it should be limited to one page.

# List of Figures and Tables

## List of Figures



## List of Tables

# List of Abbreviations and Symbols

## Abbreviations

| | |
|---|---|
| ConvNet | Convolutional neural network |
| MSE | Mean Square error |
| PSNR | Peak Signal-to-Noise ratio |

## Symbols

| | |
|---|---|
| 42 | "The Answer to the Ultimate Question of Life, the Universe, and Everything" according to [**?**] |
| $c$ | Speed of light |
| $E$ | Energy |
| $m$ | Mass |
| $\pi$ | The number pi |

# Chapter 1

# Literature Study

The first contains a general introduction to the work. The goals are defined and the modus operandi is explained. TODO: describe the problem.

## 1.1 The classical approach to speech recognition

### 1.1.1 Hidden Markov Models

### 1.1.2 Gaussian mixture models

### 1.1.3 DNN

## 1.2 Methods for Deep-Network based speech recognition

### 1.2.1 Preprocessing

#### f-bank features

f-bank (filter banks) features are one option or raw data. time domian or frequency domain?

### 1.2.2 Stochastic gradient descent

When training networks on very large training sets, working with the full data set to compute the current gradient becomes very inefficient. As a remedy its is good practice in machine learning to work with so called mini-batches. A mini-batch includes a random subset of the training data set. This procedure is known as randomized gradient descent. With an added momentum term it can be formalized as [3, page 4]:

$$\triangle w_{ij}(t) = \alpha \triangle w_{ij}(t-1) - \epsilon \frac{\partial C}{\partial w_{ij}(t)} \tag{1.1}$$

$C$ is the cost, which is computed by comparing the current network output to the desired output. $\alpha \in (0, 1)$ is a momentum coefficient. The weight of a connection

from unit $i$ in the layer under consideration to unit $j$ in the layer below is given by the expression $w_{ij}$.

### 1.2.3 Dropout

### 1.2.4 Classical layer architecture

### 1.2.5 Convnets

When looking for a signal in different parts of a recoding it is not always advisable to relearn recognition of the signal in different locations. With a conventional fully connected structure every sample on the time axis gets its own input weight. For shifted version of this signal the network will not be able to reuse weights it used to recognize the same signal at another time point in the recording. Convolutional neural nets aim to solve this problem [2, page 6], while preserving essential ordering information.

### 1.2.6 RNNs

### 1.2.7 Tensor-flow

## 1.3 Listen, Attend and Spell

The Listen Attend and Spell architecture (LAS) is the main Idea around which this thesis revolves. This section is based on [1]. The las-network consists of two mayor parts, the listener and the speller. The listener is a pyramidal recurrent neural net. It accepts filter bank spectra $\mathbf{x}_n$ as inputs and produces high level output features $\mathbf{h}_m$. The speller in turn accepts the features as input and outputs distributions over Latin character sequences $\mathbf{y}_p$. An overview of the las-achrcitecture is given in figure 1.1.

### 1.3.1 The listener

The listener shown in figure 1.1 on the bottom, consists of Bidirectional Long Short Term Memory RNN (BLSTM) blocks. These blocks are arranged in a pyramidal structure, such that the time resolution is cut in half in every layer. This operation reduces the length $U$ of the high level features $\mathbf{H}$. Without this compression the following attend and spell operation has a hard time extracting the relevant information. Additionally the compression reduces the problem complexity, which speeds up the training process significantly [1, page 4].

### 1.3.2 Attend and spell

The speller takes the features and produces a distribution over Latin character sequences as output. The computation of this output involves the context vector $\mathbf{c}_i$, the decoder state $\mathbf{s}_i$, the features $\mathbf{H}$ and the previous output $\mathbf{y}_i$. The index $i$ denotes time, $i-1$ is used to refer to results from the last time step.

These values are computed using [1, page 4]:

$$s_i = \text{RNN}(\mathbf{s}_{i-1}, \mathbf{y}_{i-1}, \mathbf{c}_{i-1}) \tag{1.2}$$

$$\mathbf{c}_i = \text{AttentionContext}(\mathbf{s}_i, \mathbf{H}) \tag{1.3}$$

$$P(\mathbf{y}_i|\mathbf{x}, \mathbf{y}_{<i}) = \text{CharacterDistribution}(s_i, \mathbf{c}_i) \tag{1.4}$$

The state follows from a recurrent neural net (RNN) made of a two layer LSTM. The attention mechanism, called AttentionContext above, computes a new context vector once every time step. This computation starts with the determination of the scalar energy $e_{i,u}$, which will be used as weight for its corresponding feature vector $h_u$. The computation starts with two feedforward neural networks or multilayer perceptrons (MLP), $\phi$ and $\psi$ [1, page 5]:

$$e_{i,u} = \phi(\mathbf{s}_i)^T \psi(\mathbf{h_u}) \tag{1.5}$$

$$\alpha_{i,u} = \frac{\exp(e_{i,u})}{\sum_u \exp(e_{i,u})} \tag{1.6}$$

$$\mathbf{c}_i = \sum_u \alpha_{i,u} \mathbf{h}_u \tag{1.7}$$

$\alpha$ is produced by running $\mathbf{e}$ trough a softmax function, which scales $\mathbf{e}$ such that all elements are within $(0, 1)$ and add up to one. These scaled weights, can then be used to form the context vector $\mathbf{c}_i$. When the training process converges the $\alpha_i$s typically follow a distribution with sharp edges[1, page 5]. Thus it is justified to think of the alphas as a sliding window. This window contains only those parts of the condensed input data set, which are currently relevant.

### 1.3.3 Training

For end-to-end speech recognition the all networks must be trained jointly. ...

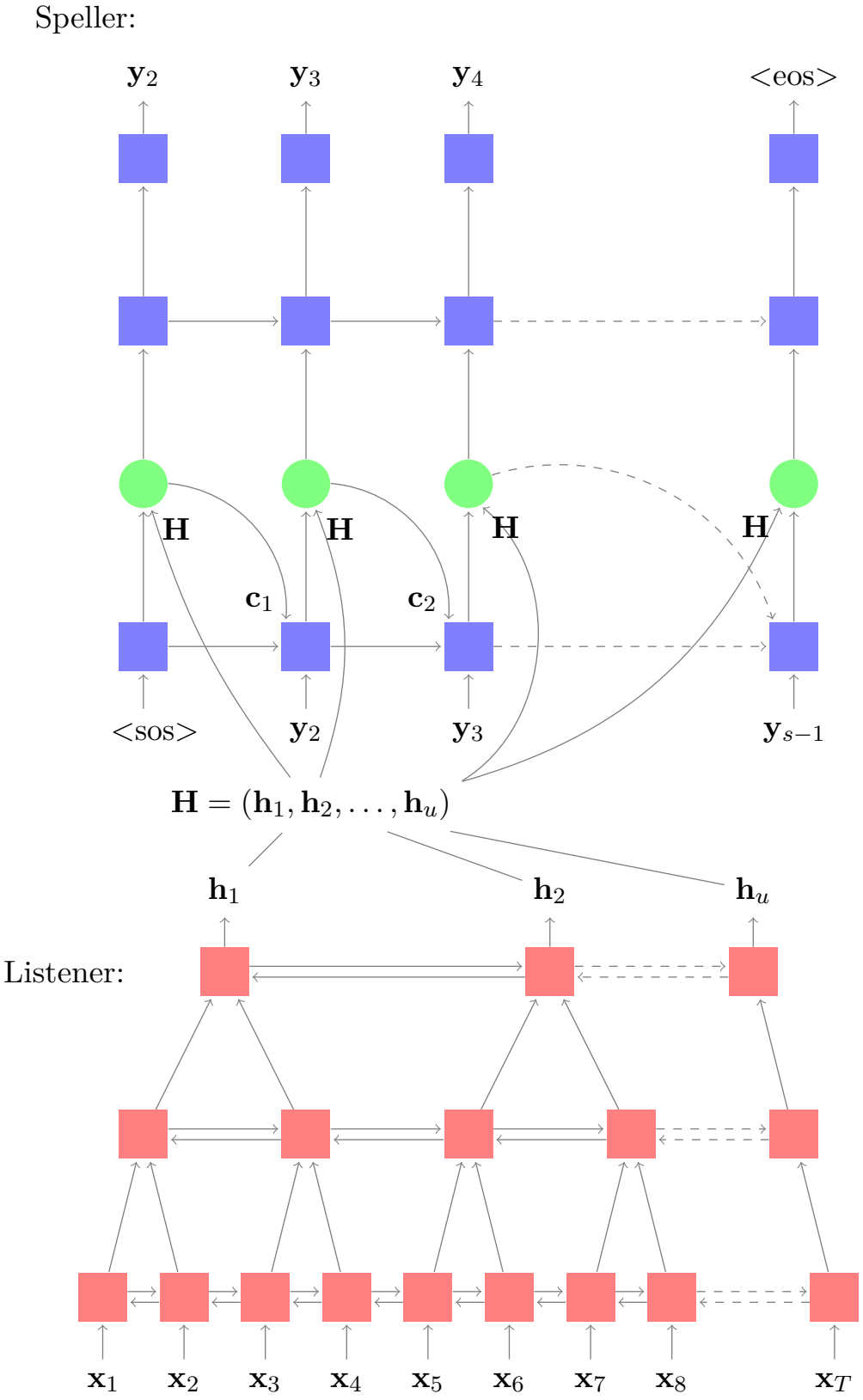


Figure 1.1: The LAS architecture [1, page 3]. BLSTM blocks are shown in red. LSTM blocks in blue and attention nets in green.

# Bibliography

[1] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals. Listen, attend and spell. *arXiv preprint*, pages 1–16, 2015.

[2] V. Dumoulin and F. Visin. A guide to convolution arithmetic for deep learning. pages 1–28, 2016.

[3] G. Hinton, L. Deng, D. Yu, G. Dahl, A.-r. Mohamed, N. Jaitly, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury. Deep Neural Networks for Acoustic Modeling in Speech Recognition. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

# Master thesis filing card

*Student*: Moritz Wolter

*Title*: Building a state of the art speech recogniser

*UDC*: 621.3

*Abstract*:

In the past machine learning relied heavily on algorithms designed by experts to solve a specific task. Which lead to highly sophisticated algorithms, which could be grasped only by small groups of people. The human brain however does not work this way, although specialized areas exist, these areas consist of similar building blocks. Artificial neural networks attempt to mimic this layout. Similar algorithmic structures are used for a wide variety of tasks. This thesis deals with the application of neural networks in speech recognition. Replacing the various subsystems by one integrated network based approach.

Thesis submitted for the degree of Master of Science in Mathematical Engineering

*Thesis supervisor*: Prof. dr. ir. Patrick Wambacq

*Assessor*: Prof. dr. ir. Johan Suykens

*Mentor*: Ir. Vincent Renkens