

# Thesis Intermediate Presentation

Moritz Wolter

August 31, 2016

# Outline

- 1 Overview
- 2 Preprocessing
- 3 Tools from Deep Learning
- 4 Listen Attend and Spell
- 5 Planning
- 6 Questions

# Project Overview

- Transcribe speech utterances to characters.
- Use a listen attend and spell (LAS) model to do this.
- Train model components jointly.

# Mel-Scale and Mel-Filter-Banks

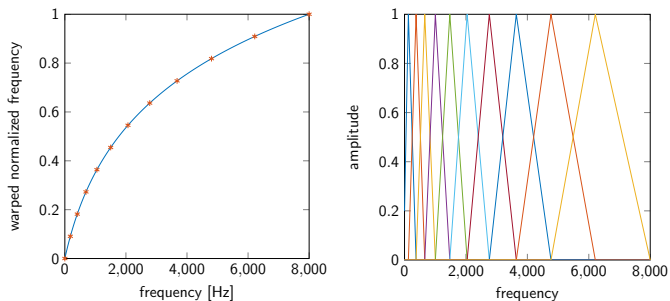


Figure 1 : The Mel scale and mel banks.

# Mel-Scale and Mel-Filter-Banks

$$B(f) = 1125 \ln(1 + f/700) \quad (1)$$

$$H_m = 0 \quad \text{if } k < f[m-1] \quad (2)$$

$$H_m = \frac{k - f[m-1]}{f[m] - f[m-1]} \quad \text{if } f[m-1] \leq k \leq f[m] \quad (3)$$

$$H_m = \frac{f[m+1] - k}{f[m+1] - f[m]} \quad \text{if } f[m] \leq k \leq f[m+1] \quad (4)$$

$$H_m = 0 \quad \text{if } k > f[m+1] \quad (5)$$

# Recurrent Neural Nets

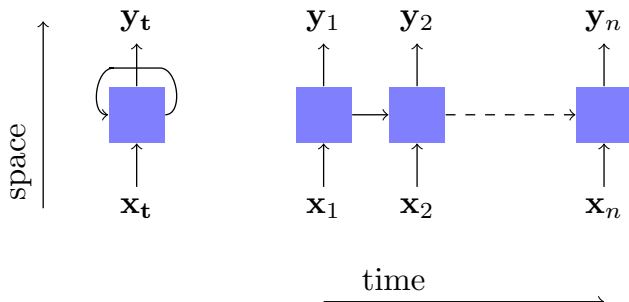


Figure 2 : Unrolling a recurrent neural net.

# Simple Cell

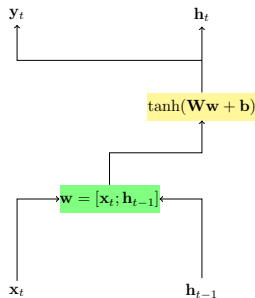


Figure 3 : Simple recurrent cell.

# Long Short Term Memory (LSTM)

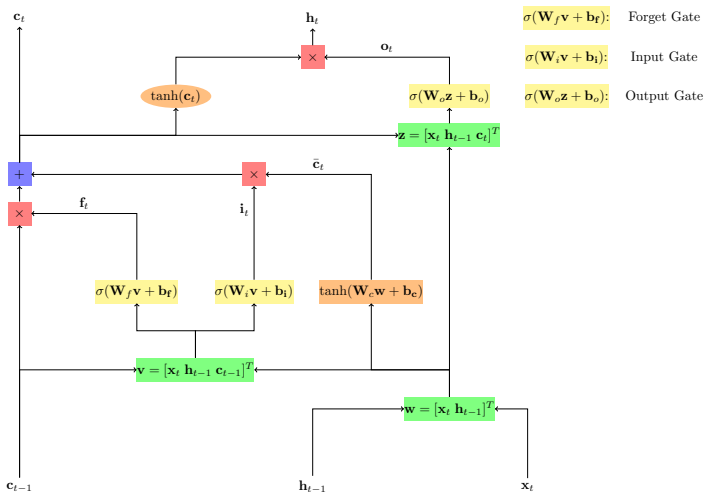


Figure 4 : Visualization of the LSTM architecture



# Long Short Term Memory (LSTM)

$$\mathbf{i}_t = \sigma(\mathbf{W}_i[\mathbf{x}_t \mathbf{h}_{t-1} \mathbf{c}_{t-1}]^T + \mathbf{b}_i) \quad (6)$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_f[\mathbf{x}_t \mathbf{h}_{t-1} \mathbf{c}_{t-1}]^T + \mathbf{b}_f) \quad (7)$$

$$\mathbf{c}_t = \mathbf{f}_t \mathbf{c}_{t-1} + \mathbf{i}_t \tanh(\mathbf{W}_c[\mathbf{x}_t \mathbf{h}_{t-1}]^T + \mathbf{b}_c) \quad (8)$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_o[\mathbf{x}_t \mathbf{h}_{t-1} \mathbf{c}_t]^T + \mathbf{b}_o) \quad (9)$$

$$\mathbf{h}_t = \mathbf{o}_t \tanh(\mathbf{c}_t) \quad (10)$$

# Bidirectional LSTM

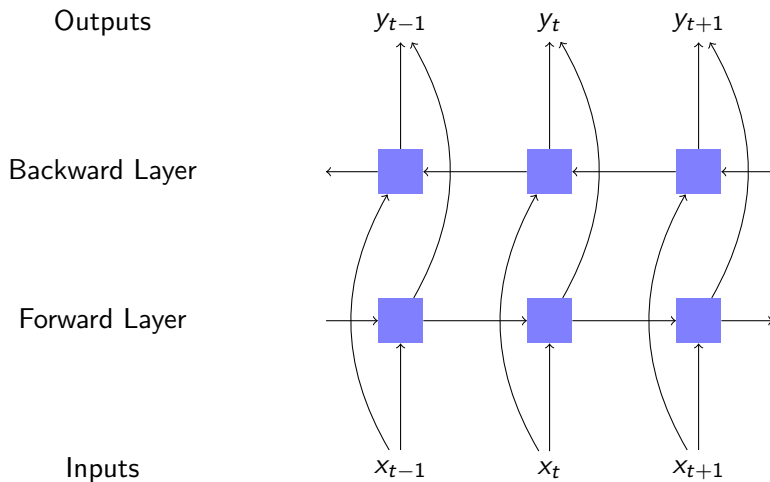


Figure 5 : BLSTM architecture

# The LAS-Architecture

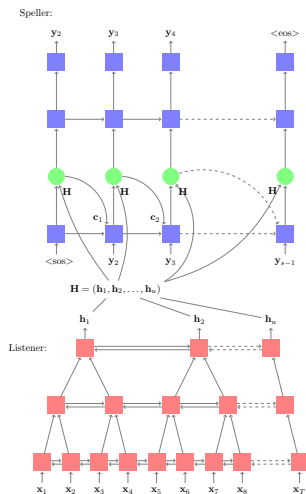


Figure 6 : The LAS architecture

# Listener

- Pyramidal Bidirectional long short term memory (*pBLSTM*).
- Pyramid structure compresses the features.
- Three *pBLSTM*s on top of a *BLSTM* layer  $\Rightarrow$  compression factor  $2^3 = 8$ .
- Pyramidal inputs concatenate the out put from previous layers:

$$\mathbf{h}_i^j = \text{pBLSTM}(\mathbf{h}_{i-1}^j, [\mathbf{h}_{2i}^{j-1}, \mathbf{h}_{2i+1}^{j-1}]) \quad (11)$$

- $i$  denotes the time step (from 0) and  $j$  the layer.

# Attend and Spell

- attention based *LSTM* transducer.
- Find the most likely character given the features and previously found letter-probabilities.

$$\mathbf{c}_i = \text{AttentionContext}(\mathbf{s}_i, \mathbf{H}) \quad (12)$$

$$\mathbf{s}_i = \text{RNN}(\mathbf{s}_{i-1}, \mathbf{y}_{i-1}, \mathbf{c}_{i-1}) \quad (13)$$

$$P(\mathbf{y}_i | \mathbf{x}, y_{<i}) = \text{CharacterDistribution}(\mathbf{s}_i, \mathbf{c}_i) \quad (14)$$

# Attention Context

- Produce a context vector  $\mathbf{c}_i$ , with alignment information.

$$e_{i,u} = \phi(\mathbf{s}_i)^T \psi(\mathbf{h}_u) \quad (15)$$

$$\alpha_{i,u} = \frac{\exp(e_{i,u})}{\sum_u \exp(e_{i,u})} \quad (16)$$

$$\mathbf{c}_i = \sum_u \alpha_{i,u} \mathbf{h}_u \quad (17)$$

- $\phi$  and  $\psi$  are feed-forward MLP networks.
- $\mathbf{s}_i$  is the decoder state.
- The  $\alpha$ s work like a sliding window.
- $U$  denotes the total number of feature vectors.

# Decoding and Rescoring

- Humans do not read character distributions.
- Left to right beam search turns distributions into text.
- Generate a tree using the  $n$  most likely characters.
- Select from the tree according to:

$$s(\mathbf{y}|\mathbf{x}) = \frac{\log P(\mathbf{y}|\mathbf{x})}{|\mathbf{y}|_c} + \lambda \log P_{LM}(\mathbf{y}) \quad (18)$$

- The first summand is the total probability found from the tree.
- The second summand is a weighted language model contribution.

# What is Tensorflow?

- “TensorFlow is an interface for expressing machine learning algorithms and an implementation for executing such algorithms. ”<sup>1</sup>
- Computations are described by directed graphs.
- Data-Tensors flow along graph edges.
- Graphs are constructed using user specified elementary operations.
- Computations are started by requesting certain values, which leads to (partial) evaluation of the graph.

---

<sup>1</sup>TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems, Abadi et al, Google Research.



# Tensorflow

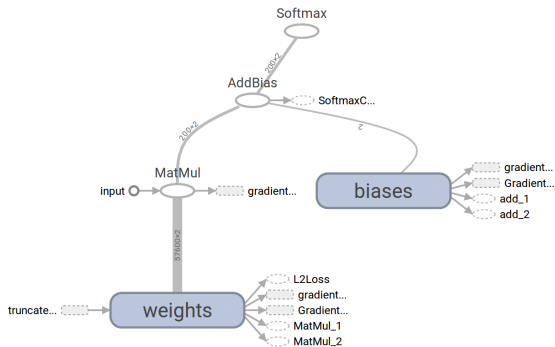


Figure 7 : A simple linear node in tensorboard

# What happend so far? What will happen next?

- What happend so far?
  - ① Literature overview.
  - ② Learned how to use tensorflow.
  - ③ Looked into timit and aurora4.
- What will happen next?
  - ① Finish implementing a skeleton LAS on Timit.
  - ② Decoding with beam search.
  - ③ Port to aurora4.

# Summary and Questions

The presentation covered:

- Input feature generation.
- The LSTM building block.
- A LAS-Architecture overview.
- The tensorflow toolbox.
- The plan.

Thank you for your attention. Questions?

`moritzalexander.wolter@student.kuleuven.be`