Thesis intermediate Presentation

Moritz Wolter

August 11, 2016

Outline

1. The Goal
2. Preprocessing
3. Deep Learning
4. Listen Attend and Spell

Speech in text out.

$$B(f) = 1125 ln(1 + f/700) \tag{1}$$

$$H_m = 0 \qquad \text{if } k < f[m-1] \tag{2}$$

$$H_m = \frac{k - f[m-1]}{f[m] - f[m-1]} \qquad \text{if } f[m-1] \le k \le f[m] \tag{3}$$

$$H_m = \frac{f[m+1] - k}{f[m+1] - f[m]} \qquad \text{if } f[m] \le k \le f[m+1] \tag{4}$$

$$H_m = 0 \qquad \text{if } k > f[m+1] \tag{5}$$

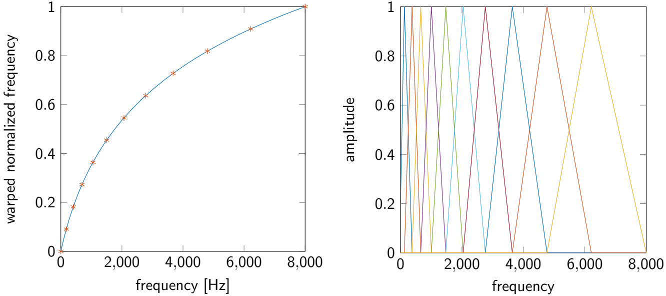


Figure 1 : The Mel scale and mel banks.

Recurrent neural nets

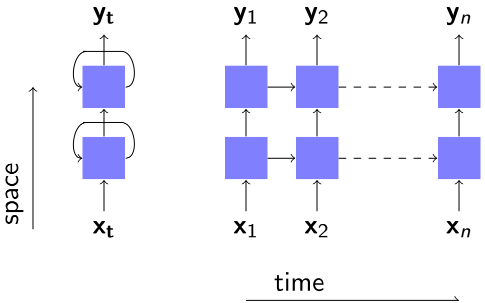


Figure 2 : Unrolling a recurrent neural net.

Gradient updates on correlated data

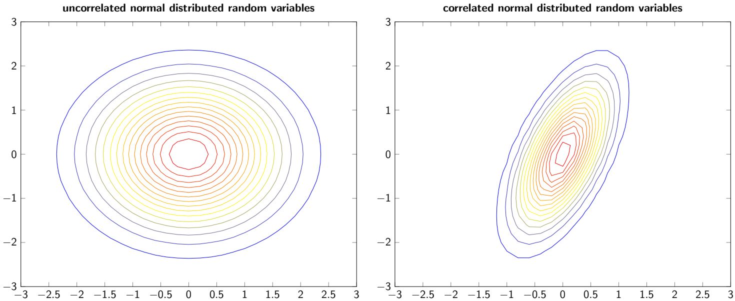


Figure 3 : Normally distributed random variables with $\Sigma_1 = \mathbf{I}$ and $\Sigma_2 = (0.25\ 0.3;\ 0.3\ 1)$.

Long Short Term Memory (LSTM)

$$\mathbf{i_t} = \sigma(\mathbf{W}_i[\mathbf{x}_t\ \mathbf{h_{t-1}}\ \mathbf{c_{t-1}}]^T + \mathbf{b}_i) \tag{6}$$

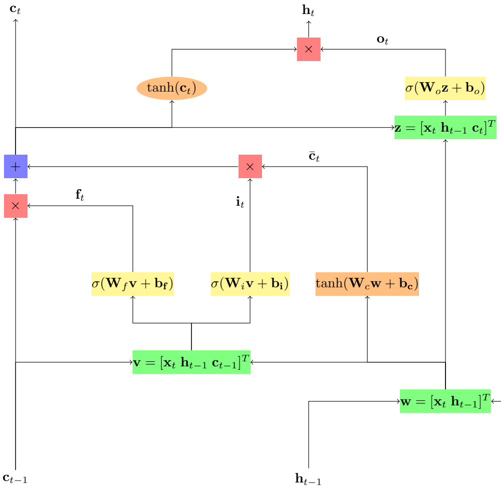
$$\mathbf{f_t} = \sigma(\mathbf{W}_f[\mathbf{x}_t\ \mathbf{h_{t-1}}\ \mathbf{c_{t-1}}]^T + \mathbf{b}_f) \tag{7}$$

$$\mathbf{c_t} = \mathbf{f}_t\mathbf{c_{t-1}} + \mathbf{i}_t \tanh(\mathbf{W}_c[\mathbf{x}_t\ \mathbf{h_{t-1}}]^T + \mathbf{b}_c) \tag{8}$$

$$\mathbf{o_t} = \sigma(\mathbf{W}_o[\mathbf{x}_t\ \mathbf{h_{t-1}}\ \mathbf{c_t}]^T + \mathbf{b}_o) \tag{9}$$

$$\mathbf{h_t} = \mathbf{o_t} \tanh(\mathbf{c_t}) \tag{10}$$

Long Short Term Memory (LSTM)


$\mathbf{c}_t$
$\mathbf{h}_t$
$\times$
$\mathbf{o}_t$
$\tanh(\mathbf{c}_t)$
$\sigma(\mathbf{W}_o\mathbf{z}+\mathbf{b}_o)$
$\mathbf{z}=[\mathbf{x}_t\ \mathbf{h}_{t-1}\ \mathbf{c}_t]^T$
$+$
$\times$
$\bar{\mathbf{c}}_t$
$\mathbf{f}_t$
$\times$
$\mathbf{i}_t$
$\sigma(\mathbf{W}_f\mathbf{v}+\mathbf{b_f})$
$\sigma(\mathbf{W}_i\mathbf{v}+\mathbf{b_i})$
$\tanh(\mathbf{W}_c\mathbf{w}+\mathbf{b_c})$
$\mathbf{v}=[\mathbf{x}_t\ \mathbf{h}_{t-1}\ \mathbf{c}_{t-1}]^T$
$\mathbf{w}=[\mathbf{x}_t\ \mathbf{h}_{t-1}]^T$
$\mathbf{c}_{t-1}$
$\mathbf{h}_{t-1}$
$\mathbf{x}_t$

$\sigma(\mathbf{W}_f\mathbf{v}+\mathbf{b_f})$:	Forget Gate
$\sigma(\mathbf{W}_i\mathbf{v}+\mathbf{b_i})$:	Input Gate
$\sigma(\mathbf{W}_o\mathbf{z}+\mathbf{b}_o)$:	Output Gate

Figure 4 : Visualization of the LSTM architecture

Bidirectional BLSTM

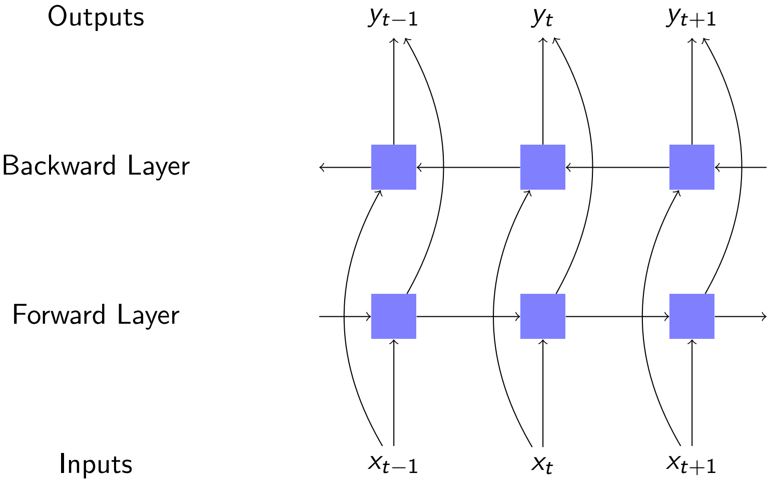


Figure 5 : Bidirectional LSTM architecture

Listener

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

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

- 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 letters.

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

$$\mathbf{s}_i = \text{RNN}(\mathbf{s}_i, \mathbf{H}) \tag{13}$$

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

Attention Context

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

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

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

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

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

The LAS-Architecture

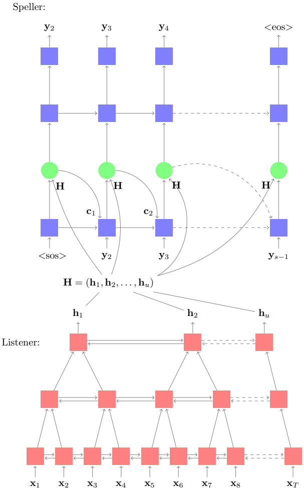


Figure 6 : The LAS architecture

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:

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

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

What is tensorflow

A machine learning toolbox.

Tensorflow

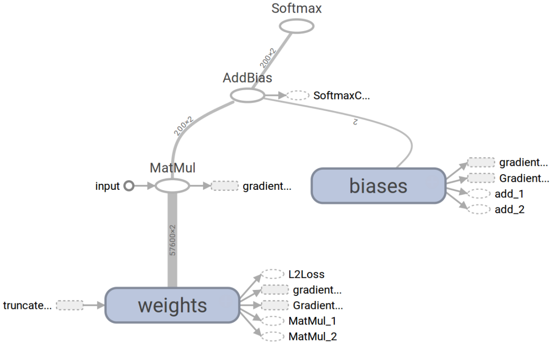


Figure 7 : A simple linear node in tensorboard

Summary and questions