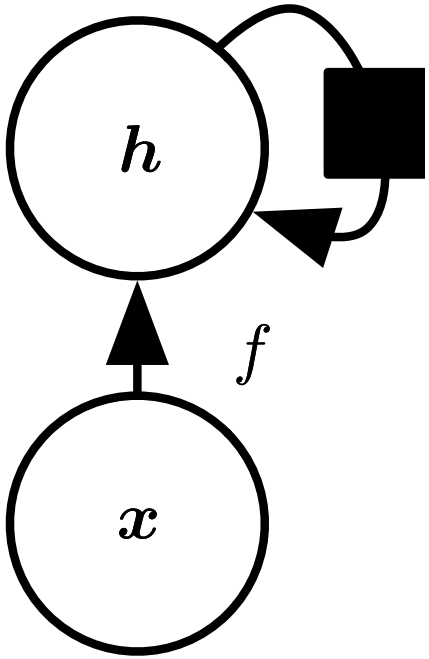
Recurrent Neural Networks

Oliver Dürr

Datalab-Lunch Seminar Series
Winterthur, 10. Januar 2017

Code: github.com/oduerr/dl_tutorial/



Outline

- Use cases (for Kurt ;-))
- Introduction
- Toy Example
  - Forward pass
  - Training in TensorFlow
- Different Architectures
  - Deep RNNs, Bidirectional
- Vanishing Gradient Problem
  - Flow of Gradient in Computational graph
- LSTM
  - Gradients survive
- Fun example Character Level RNNs
- Pointers to further applications

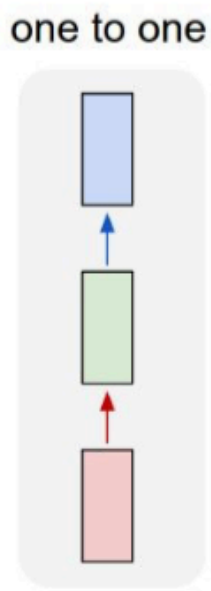
Resources

- Many figures are taken from the following resources:
  - **Deep Learning Book** chap10
    - http://www.deeplearningbook.org/contents/rnn.html
  - CS231n
    - Lecture on RNN: http://cs231n.stanford.edu/slides/winter1516_lecture10.pdf
    - Video to CS231n https://www.youtube.com/watch?v=iX5V1WpxxkY
  - Blog Posts
    - Karpathy, May 2015: The unreasonable effectiveness of Recurrent Neural Networks http://karpathy.github.io/2015/05/21/rnn-effectiveness/
    - Colah, August 2015: Understanding LSTM Networks http://colah.github.io/posts/2015-08-Understanding-LSTMs/
    - R2RT, July 2016: http://r2rt.com/recurrent-neural-networks-in-tensorflow-i.html
    - WildML, August 2016: Praktical consideration e.g. how to use sequences with different length. http://www.wildml.com/2016/08/rnns-in-tensorflow-a-practical-guide-and-undocumented-features/

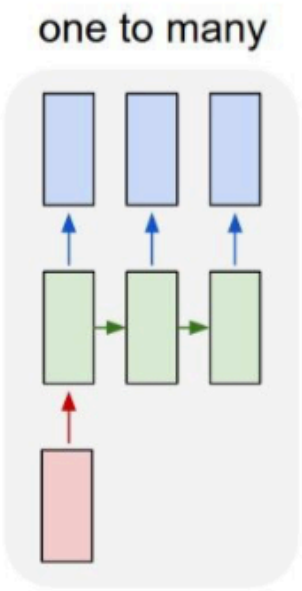
- Further ipython notebooks:
  - https://github.com/oduerr/dl_tutorial/blob/master/tensorflow/RNN

Use cases

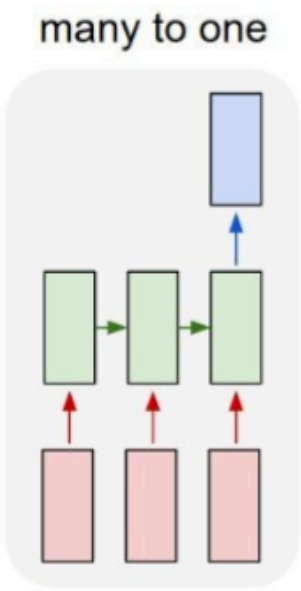
Recurrent neural networks (RNN) are used to model sequences.



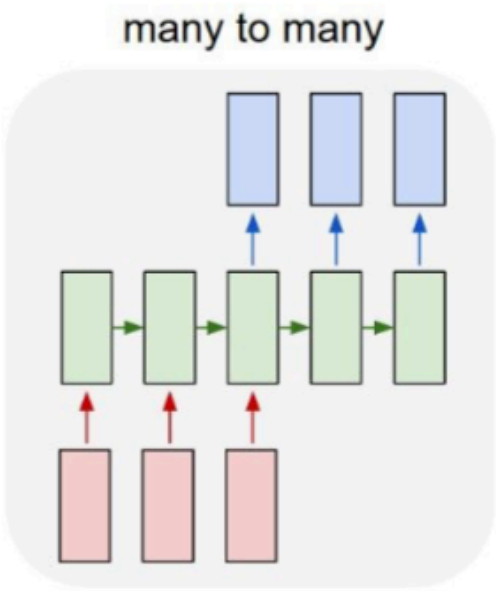
Fixed input / output.
Standard neural network no RNN



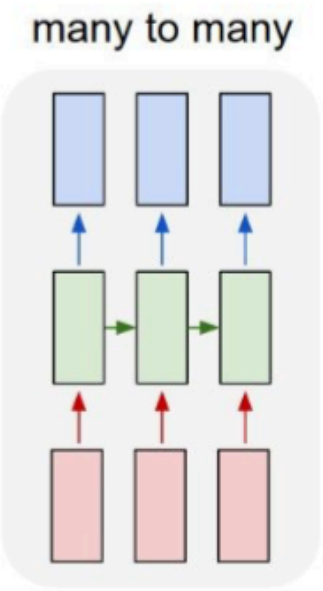
E.g. Image Captioning.
Image -> Seq of words



E.g. Sentiment Classification.
Seq of words
→Sentiment



E.g. Translation.
seq of words (french)
→seq of words (german)



E.g. Language models.
seq of letters
→seq of letters

Predicting the next letter

Illustration: http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Principle Idea: dynamical system

State (e.g. vector *s*) changes over **discrete time**, with a function *f*

$$\boldsymbol{s}^{(t)} = f(\boldsymbol{s}^{(t-1)}; \boldsymbol{\theta})$$

The function f and the parameters θ **are the same at all time points**. System is defined when θ and a state, e.g. $s^{(t=1)}$ is known.

Examples:

- Markov-Process: f is matrix, s probability vector
- Physical System in discrete time with no external forces
  - *s* (position, velocity)
- Number of fish in a lake each springtime (*s* scalar value)

Properties:

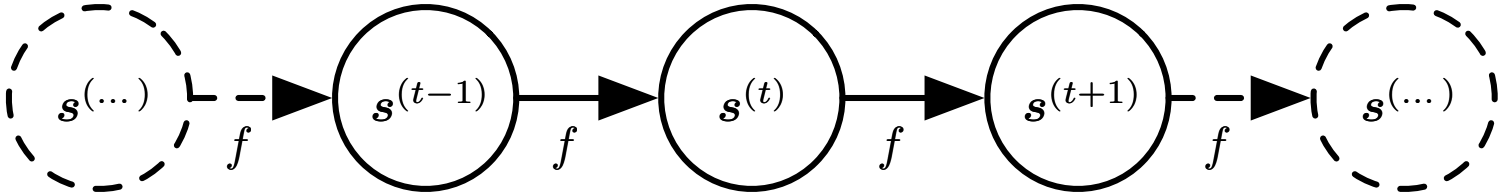
- Time-invariant absolute time does not matter for long enough sequences
- Markov property
- Works for arbitrary sequence length

Principle Idea: unfolding in time

$$s^{(t)} = f(s^{(t-1)}; \theta)$$

$$\begin{aligned} s^{(3)} &= f(s^{(2)}; \theta) \\ &= f(f(s^{(1)}; \theta); \theta) \end{aligned}$$

Written as a computational graph



Note that sometimes the computational graph, are written differently (e.g. in TensorFlow, ops are nodes)

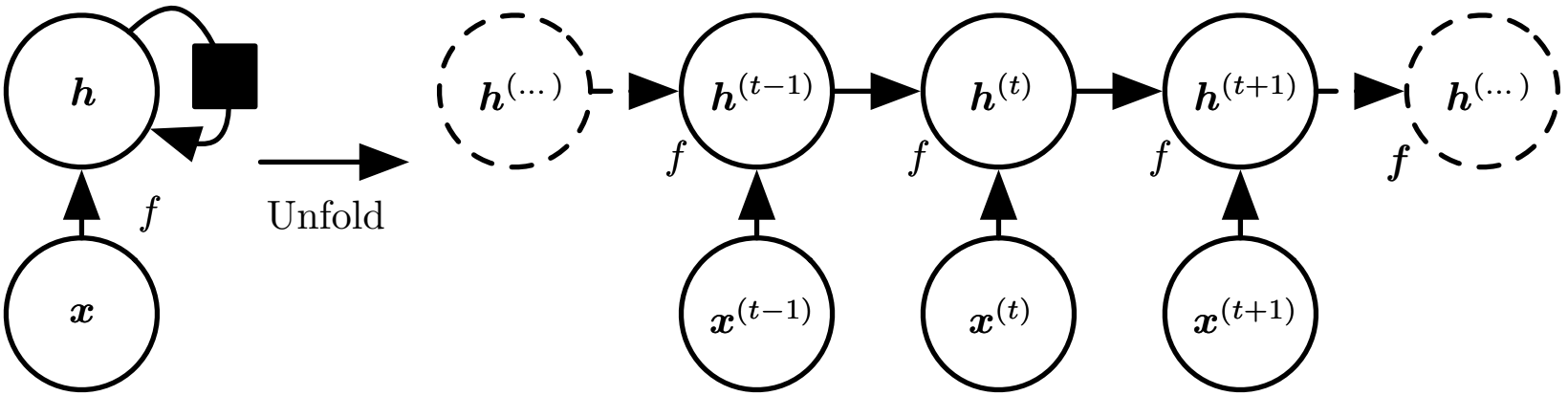
Illustration: http://www.deeplearningbook.org/

Typical RNNs are externally driven

Network is driven by sequence x(t) (a vector)

$$\boldsymbol{h}^{(t)} = f(\boldsymbol{h}^{(t-1)}, \boldsymbol{x}^{(t)}; \boldsymbol{\theta})$$

$h^{(t)}$ summarizes $(x^{(1)},\ldots,x^{(t-1)})$



Left: Circuit Diagram (black square delay of one time step)
Right: Unrolled / unfolded

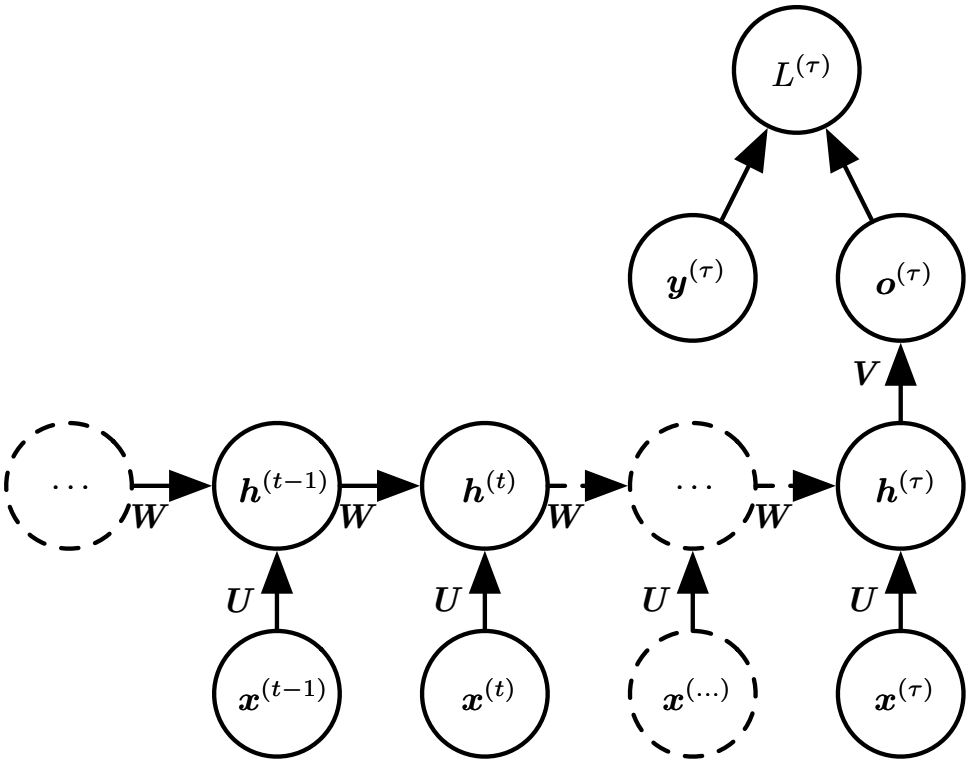
State, is now called hidden unit.
Networks usually read out (some or all) hidden units to make predictions. Various possibilities…

Illustration: http://www.deeplearningbook.org/

Example 1

A sequence $x^{(t)}$ corresponds to a single outcome y

- x e.g. words of a sentence coded with word embedding of size 300
- y vector indicating sentiment: (1,0,0) positive, (0,1,0) neutral, (0,0,1) neg.



Single loss at the end

$$\boldsymbol{o}^{(t)} = \boldsymbol{c} + \boldsymbol{V}\boldsymbol{h}^{(t)}$$

$$\boldsymbol{a}^{(t)} = \boldsymbol{b} + \boldsymbol{W}\boldsymbol{h}^{(t-1)} + \boldsymbol{U}\boldsymbol{x}^{(t)}$$

$$\boldsymbol{h}^{(t)} = \tanh(\boldsymbol{a}^{(t)})$$

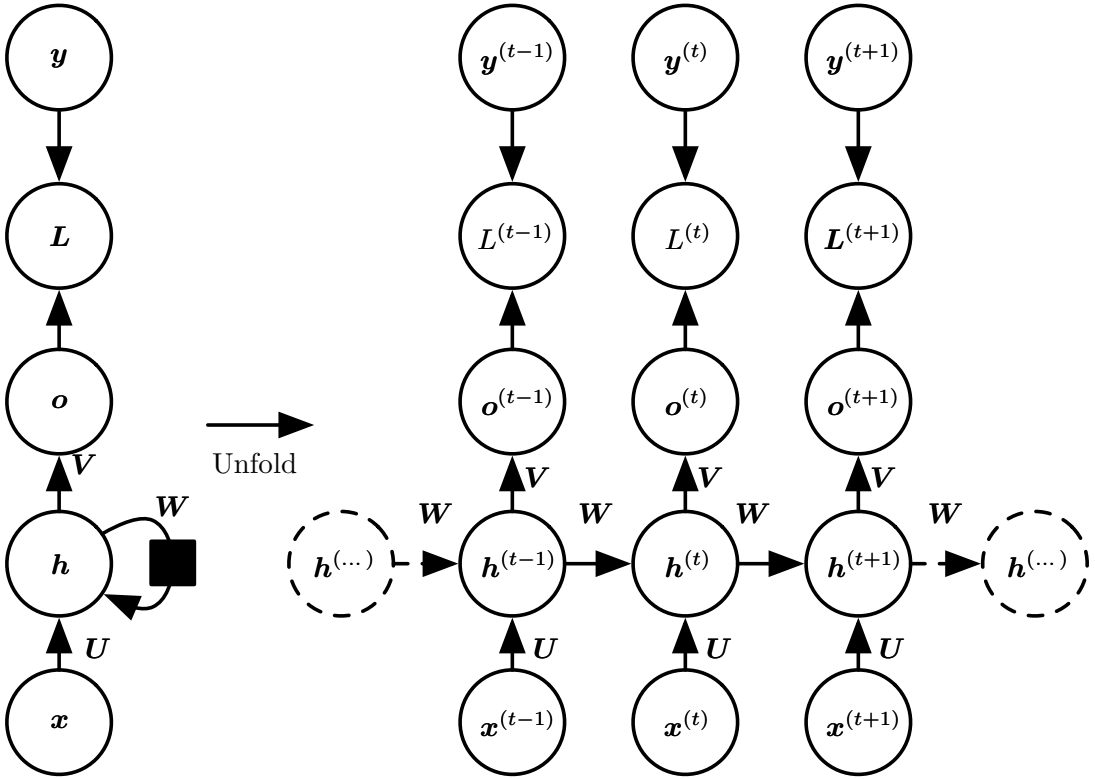
Comp. of state h in [-1,1]
W,V,U,b,c are learnt

Illustration: http://www.deeplearningbook.org/

Example 2

A sequence $x^{(t)}$ corresponds to an outcome at each time step outcomes y

- x letter in a string of letters
- y next letter



For categorical and one hot

$$L^{(t)} = y^{(t)} \cdot \log(\hat{y}^{(t)})$$

$$L = \sum_t L^{(t)}$$

$$\hat{\boldsymbol{y}}^{(t)} = \text{softmax}(\boldsymbol{o}^{(t)})$$

$$\boldsymbol{o}^{(t)} = \boldsymbol{c} + \boldsymbol{V}\boldsymbol{h}^{(t)}$$

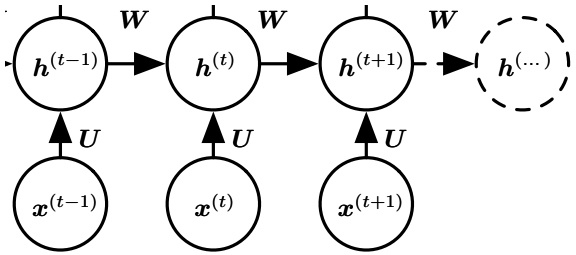
$$\boldsymbol{a}^{(t)} = \boldsymbol{b} + \boldsymbol{W}\boldsymbol{h}^{(t-1)} + \boldsymbol{U}\boldsymbol{x}^{(t)}$$

$$\boldsymbol{h}^{(t)} = \tanh(\boldsymbol{a}^{(t)})$$

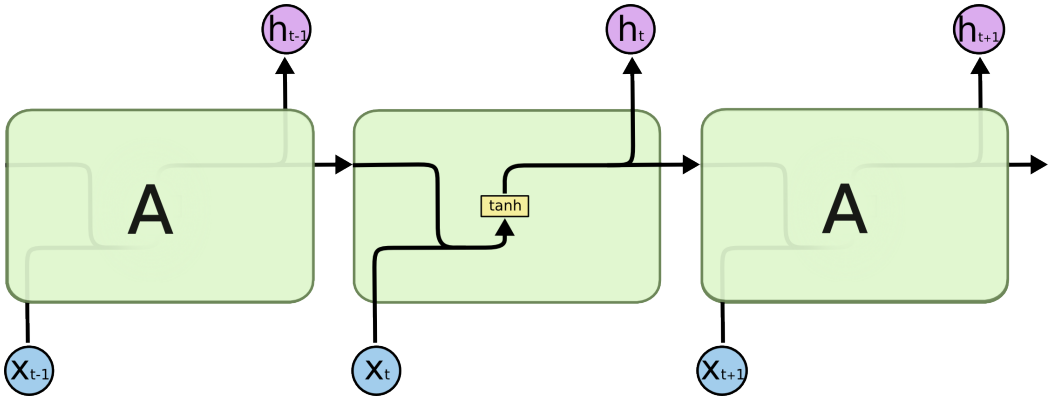
W,V,U,b,c are learnt

Illustration: http://www.deeplearningbook.org/

Alternative ways of writing RNNs



$$\boldsymbol{a}^{(t)} = \boldsymbol{b} + \boldsymbol{W}\boldsymbol{h}^{(t-1)} + \boldsymbol{U}\boldsymbol{x}^{(t)}$$
$$\boldsymbol{h}^{(t)} = \tanh(\boldsymbol{a}^{(t)})$$



Transposing:
W x → x' W'

x, h are now row vectors

$$h_t = \tanh([h_{t-1}, x_t] \cdot W + b) = \tanh(h_{t-1} \cdot W_h + x_t \cdot U + b)$$

Appending columns at vector

$$W = \begin{pmatrix} W_h \\ U \end{pmatrix}$$

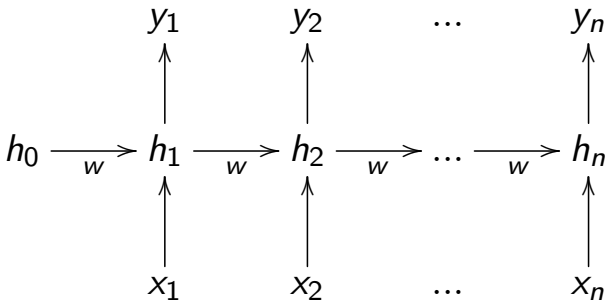
Illustration: http://www.deeplearningbook.org/ http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Feedforward vs. CNN vs. Recurrent Network

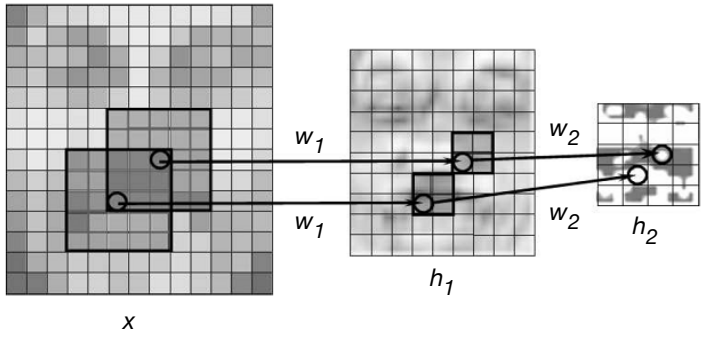
- Feedforward: No weighty sharing

Weight Sharing

Recurrent neural network shares weights between time-steps



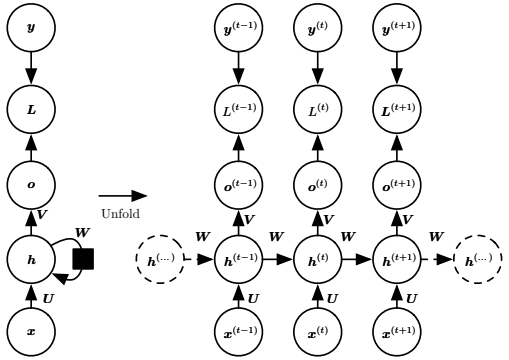
Convolutional neural network shares weights between local regions



Slide Credit: David Sliver

Example: The ice cream store

Toy Example: Ice Cream Store



y store can sell icecream

 (1,0) ice cream available

 (0,1) store has run out of ice cream

x weather  (1,0,0)  (0,1,0)  (0,0,1)

Sample

Complicated order policy we don't know

Days

https://github.com/oduerr/dl_tutorial/blob/master/tensorflow/RNN/simple_rnn.ipynb

Code walkthrough

- Forward pass in numpy
- Blackboard creation of mini-batches
- If time permits:
  - TF-implementation

Other architectures: Bidirectional RNN

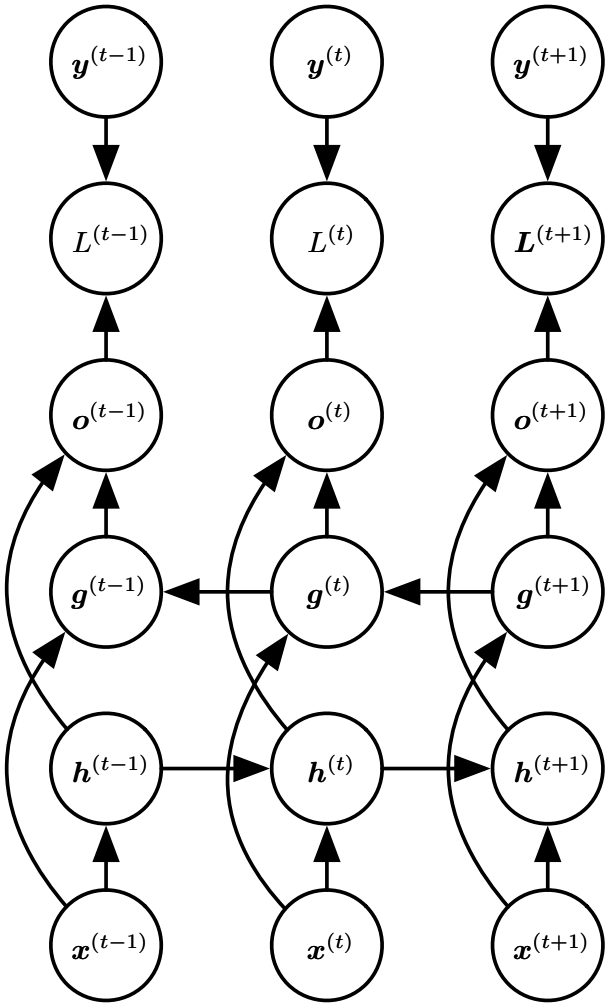
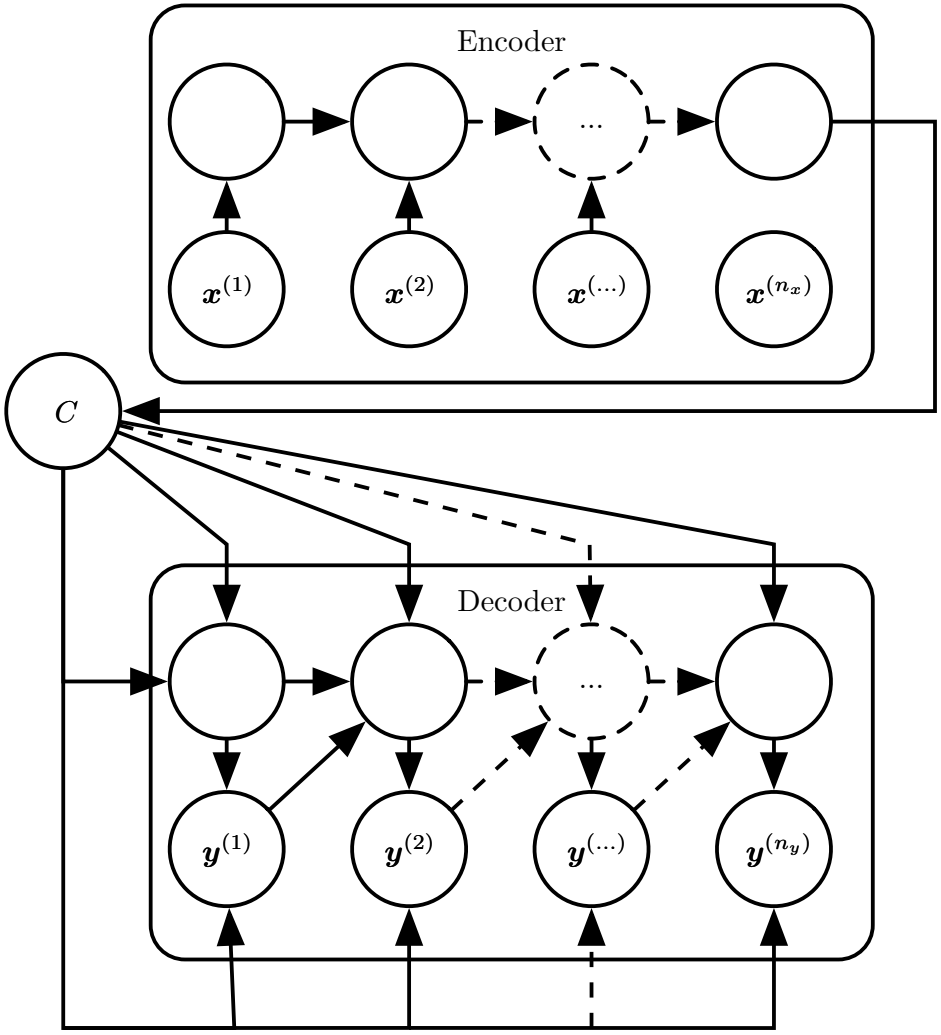


Illustration: http://www.deeplearningbook.org/

Other architectures: Sequence to Sequence



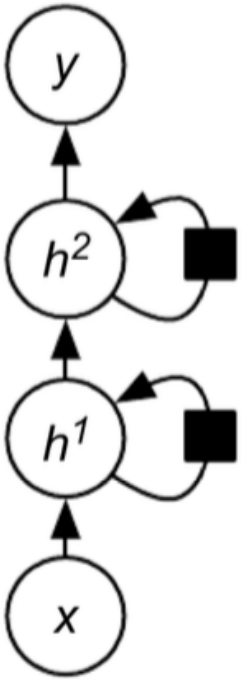
Encoder Language 1

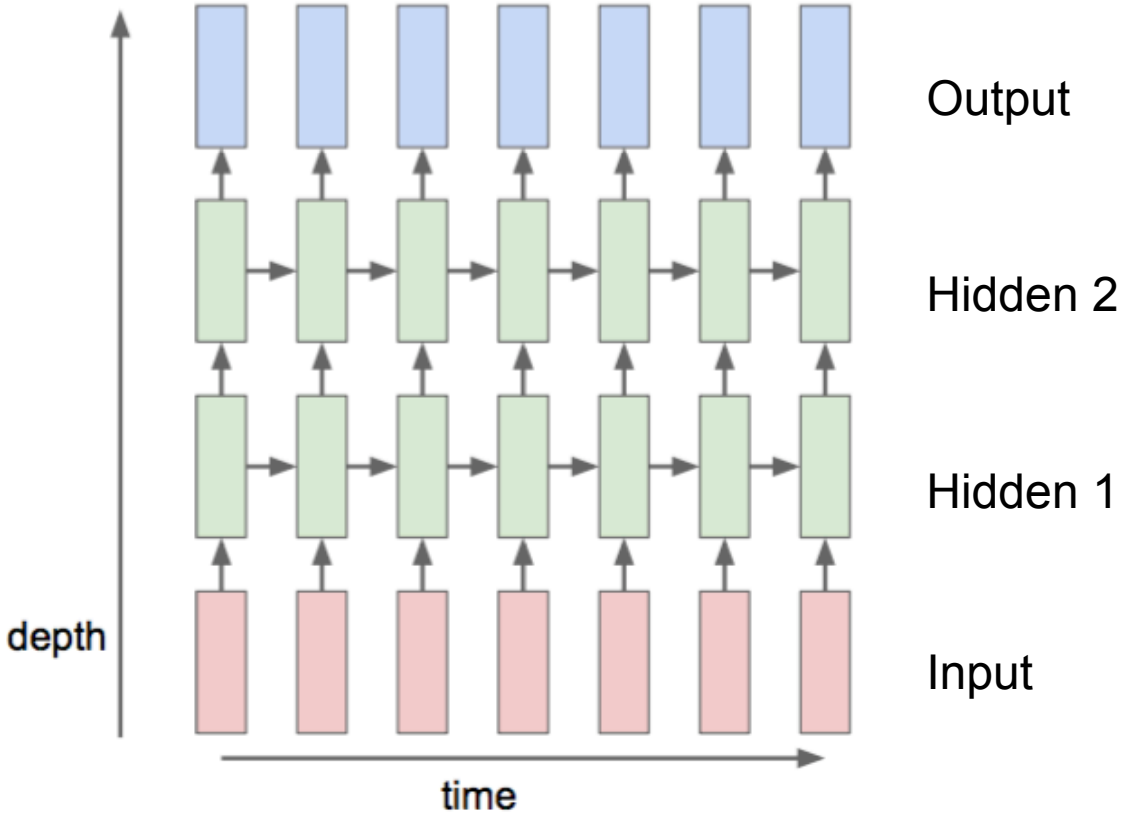
All information is in the context C
Hinton: “Thought Vector”

Special design of decoder
‘output feedback’

Illustration: http://www.deeplearningbook.org/

Other architectures: Deep RNNs





Simply use the output h as a new input. Other approaches are possible, see e.g. DL-book

Illustration: http://www.deeplearningbook.org/ and : http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Vanishing Gradient

Vanishing Gradient

- Long range dependencies can be found for many systems and are important to model
- Long range interactions cannot be trained with standard RNN
  - Vanishing Gradient (Hochreiter 1991, Diplomarbeit “Untersuchungen zu dynamischen neuronalen Netzen”)
- Let’s have a look who to calculated gradients in computational graphs
  - Interesting and import by it’s own right!

Gradient flow in a computational graph

How is h effected by x?

$$h(z) = h(g(f(x)))$$

$$\frac{\partial h}{\partial x} = \frac{\partial y}{\partial x}\frac{\partial z}{\partial y}\frac{\partial h}{\partial z}$$

$$\frac{\partial h}{\partial x} = \frac{\partial f(x)}{\partial x}\frac{\partial g(y)}{\partial y}\frac{\partial h(z)}{\partial z}$$

$$\frac{\partial h}{\partial x} = \frac{\partial f(x)}{\partial x}\frac{\partial h(z)}{\partial y}$$

How is h effected by y?

$$h(z) = h(g(y))$$

$$\frac{\partial h}{\partial y} = \frac{\partial z}{\partial y}\frac{\partial h}{\partial z}$$

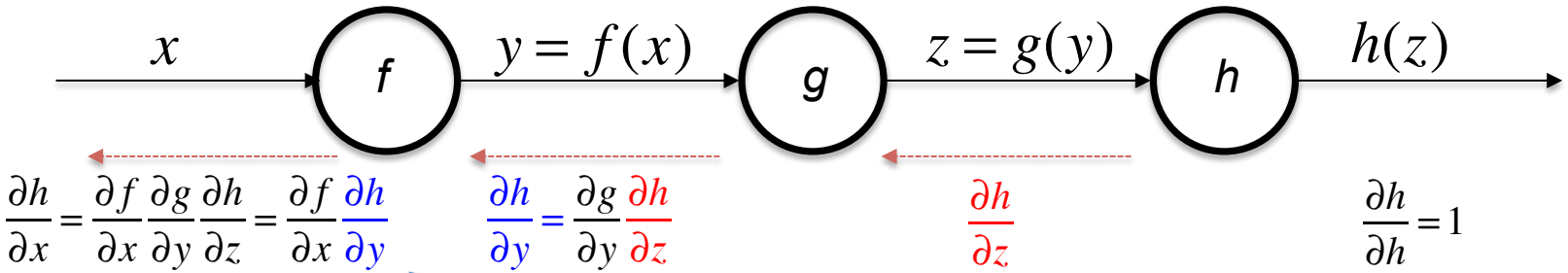
$$\frac{\partial h}{\partial y} = \frac{\partial g(y)}{\partial y}\frac{\partial h(z)}{\partial z}$$

How is h effected by z?

$$h(z) = h(z)$$

$$\frac{\partial h}{\partial z}$$

For DL: h is Loss



incoming gradient

outgoing gradient of an operation f = $\frac{\partial f}{\partial x}$ • incoming gradient

Gradient flow in a computational graph: local junction

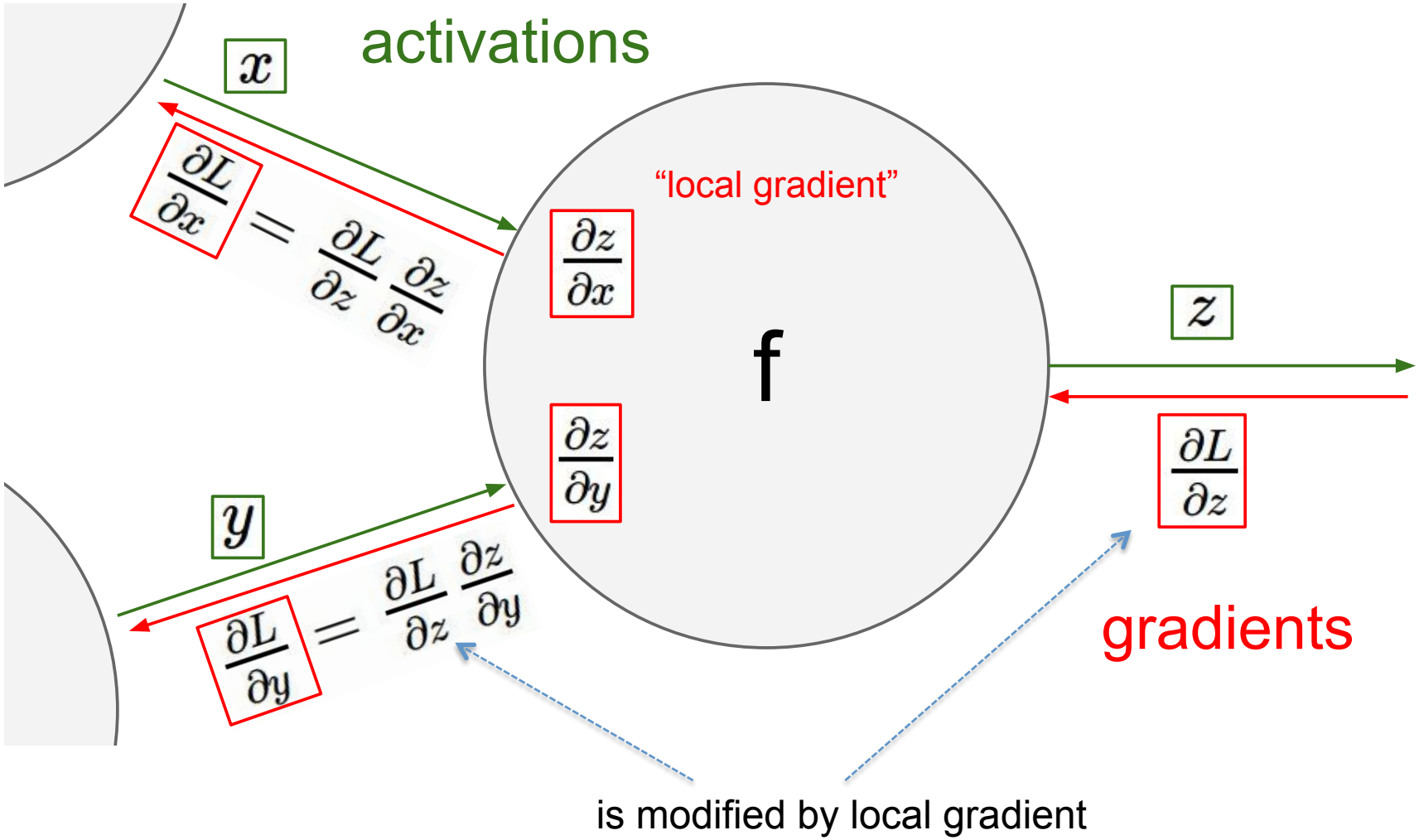
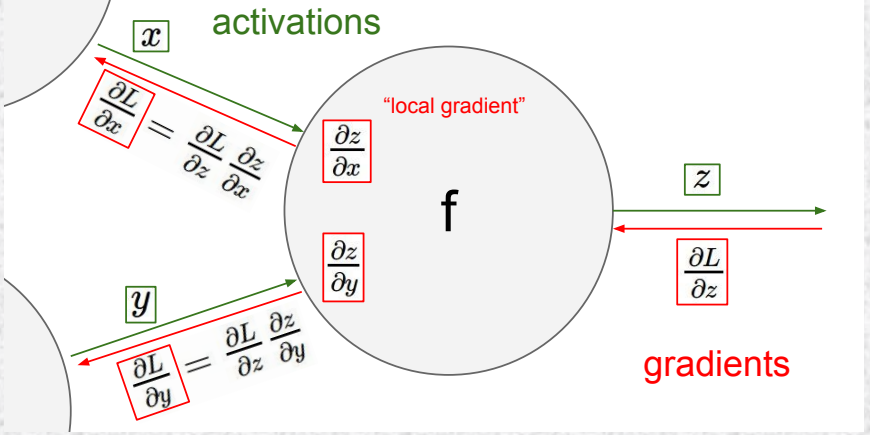


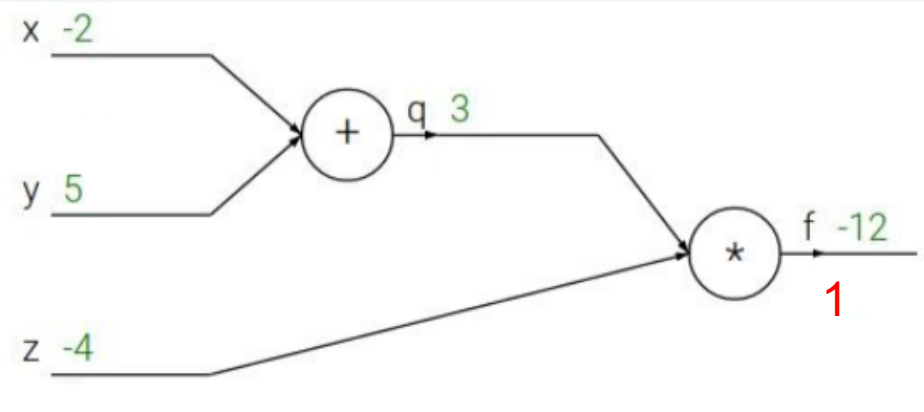
Illustration: http://cs231n.stanford.edu/slides/winter1516_lecture4.pdf

Example

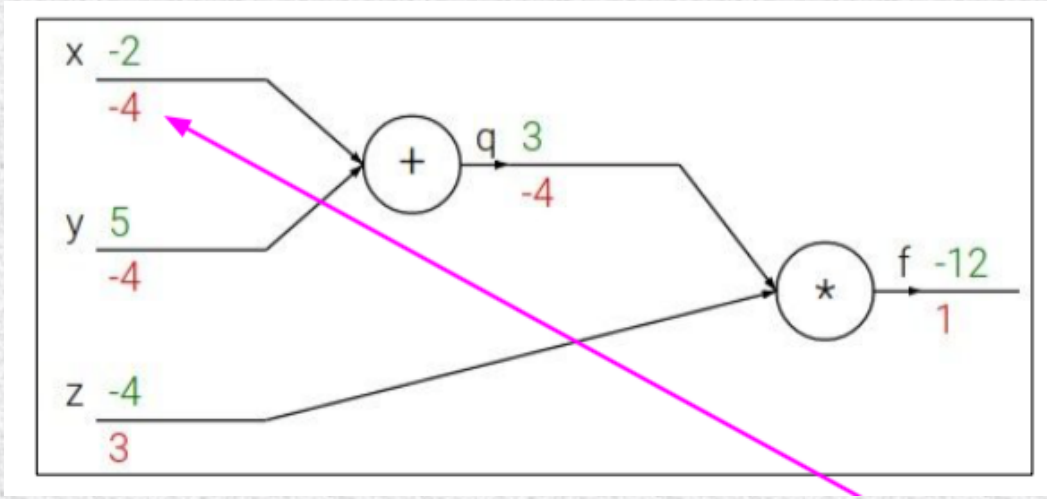
$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4



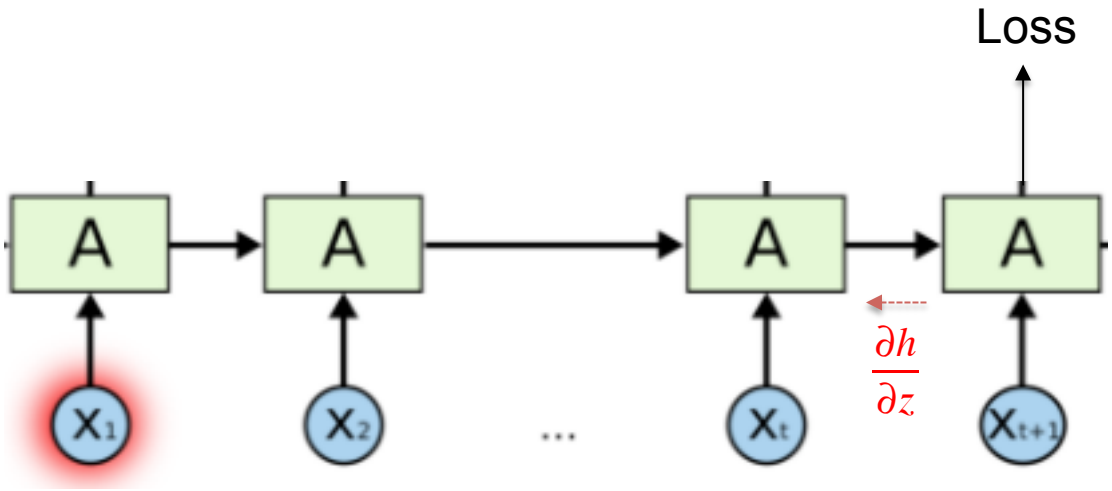


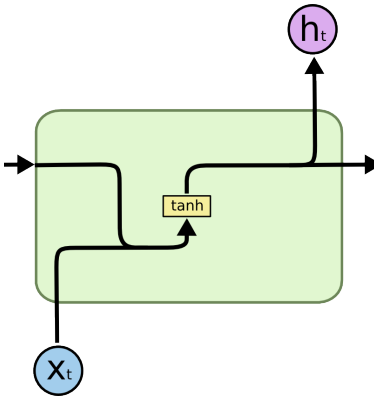
$$\frac{\partial(\alpha+\beta)}{\partial\alpha} = 1 \qquad \frac{\partial(\alpha*\beta)}{\partial\alpha} = \beta$$



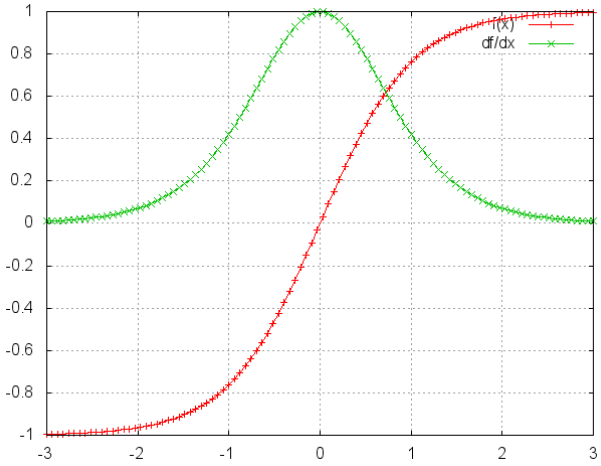
➔Multiplication do a switch

Issues with simple RNN (hand waving)





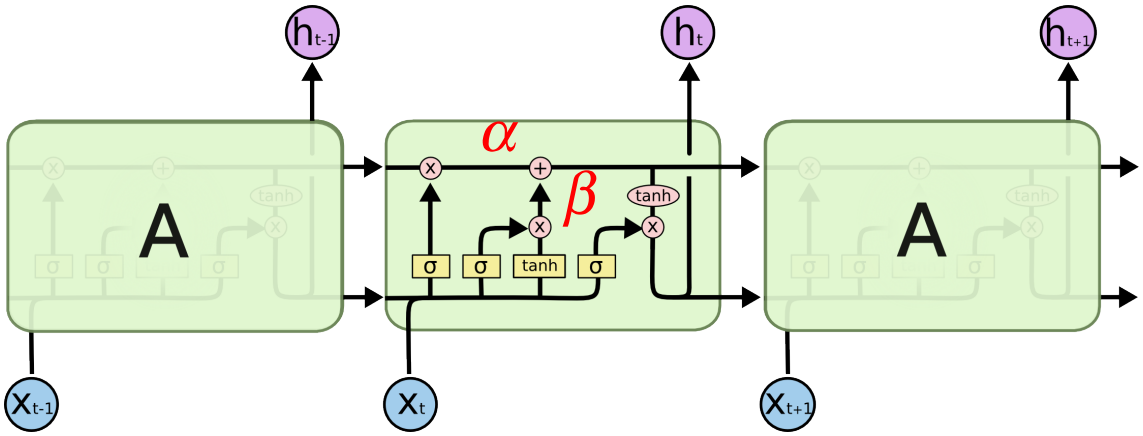
Repeated: Multiplication of grad tanh which is < 1

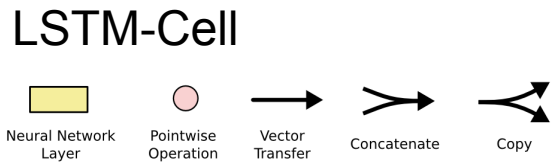


A bit more complicated since weights are shared and are matrices

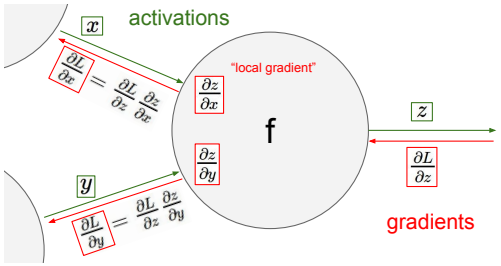
Illustration: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Remedy Long Short Term Memory (LSTM) Cells





Forget about the forget gate (may be switched off in the beginning of training)



$$\frac{\partial(\alpha+\beta)}{\partial\alpha}=1$$

➔'Gradient Super Highways' on the upper conveyor belt C_t

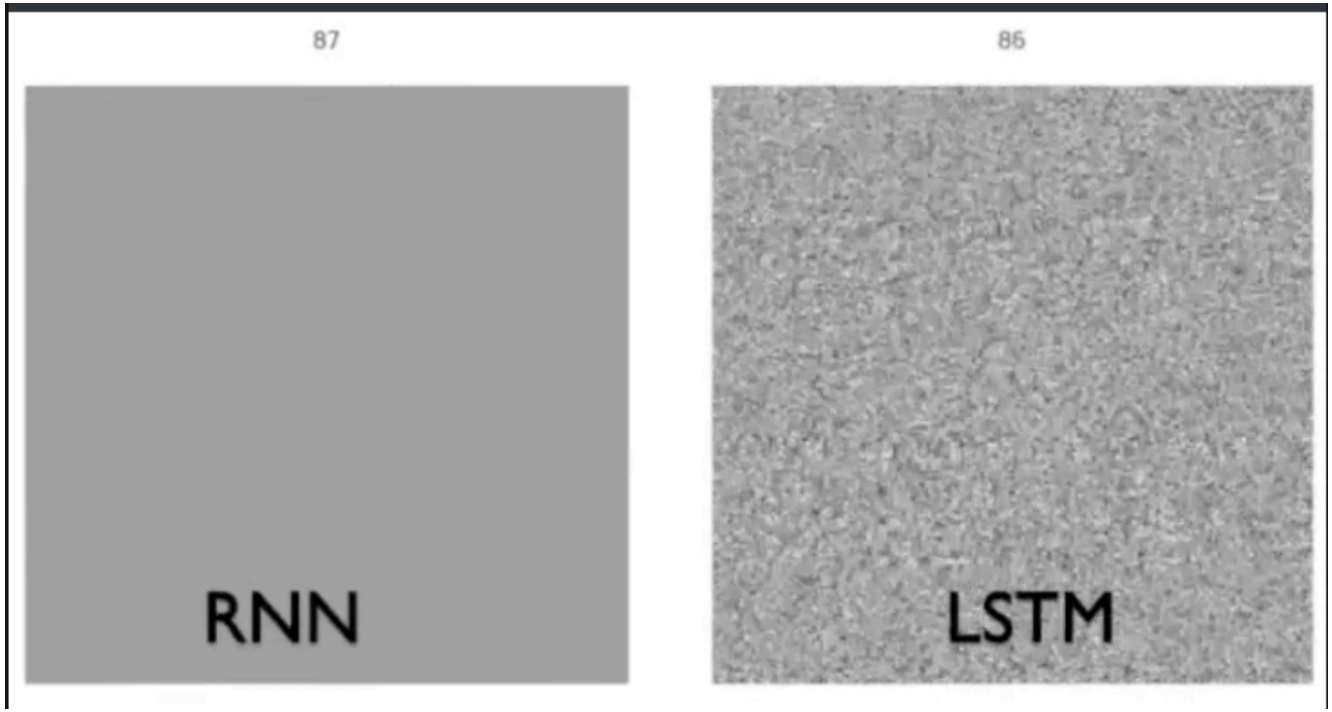
What comes in (on the right) does go out (on the left)

Similar to high way networks or resnets

Illustration: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

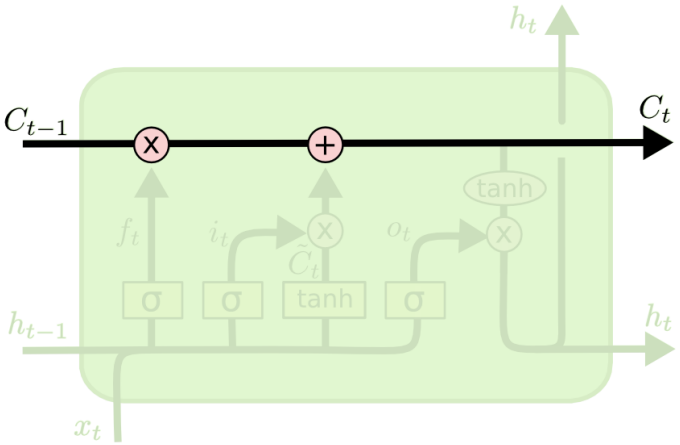
"Experimental Verification"

http://imgur.com/gallery/vaNahKE



Intuition of the LSTM-Cells

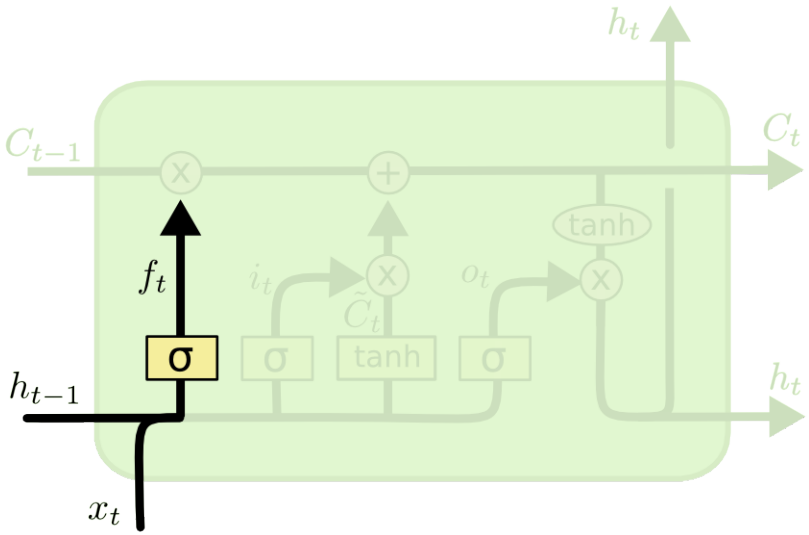
- We consider the cell state C_t to code information like gender



- Example, for Karaparthys post.

- Value of a particular component of C_t green=1, blue = -1, seems to code url
- Not all components of C_t are so easily interpretable

Run through LSTM (forget gate)



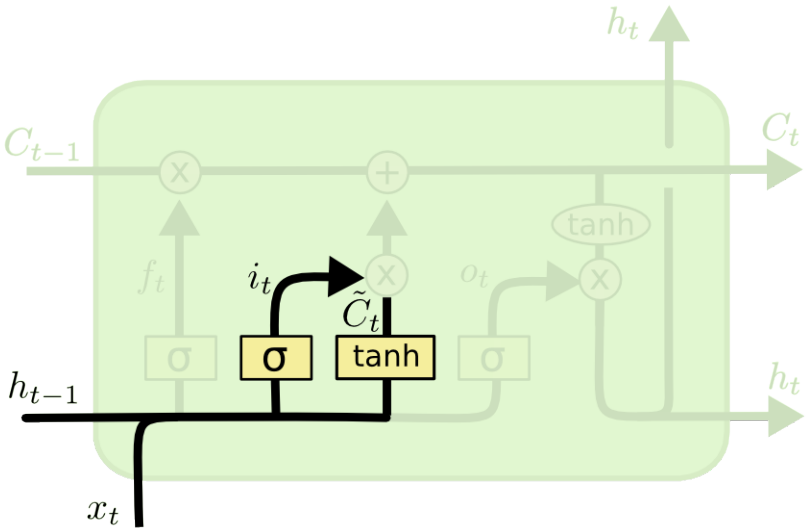
$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

Forget gate looks at previous h-state h_t-1 and input x_t determines the cell state should be forgotten. “Example is Gender State still important?”

Example: f_t = (1,1,0,1,1) would forget the state 3 (binary for illustration). Gender is not important anymore

Taken from Colah’s Blog: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Run through LSTM (proposing new addition)



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$

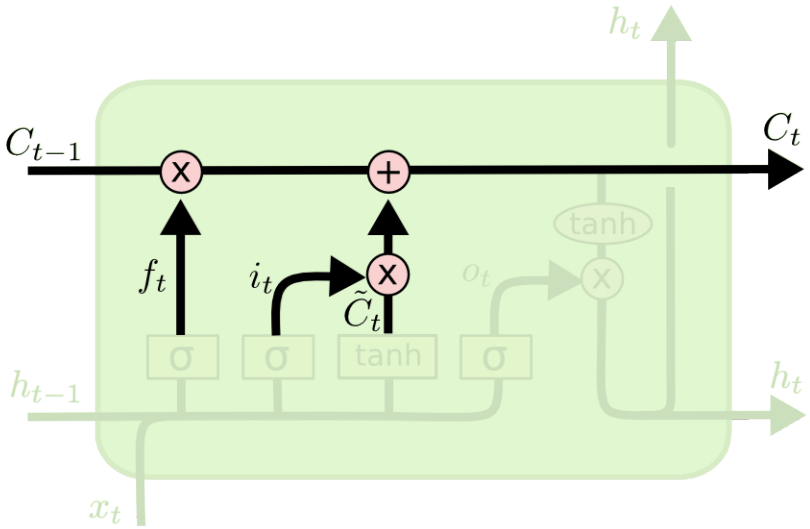
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

This part looks at previous h-state h_t-1 and input x_t and determines a candidate for the new cell state (e.g. new gender).

Example: \tilde{C_t} = (0,0,-1,0,0) would set state 3 to -1

Taken from Colah's Blog: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Run through LSTM (adding to the new state)

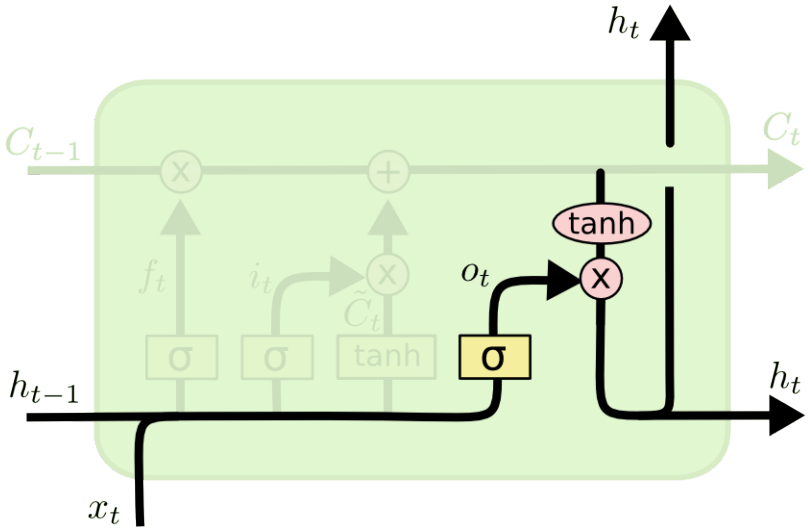


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Adds the new cell state (e.g. new gender).

Taken from Colah's Blog: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Run through LSTM (determining output)



$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$

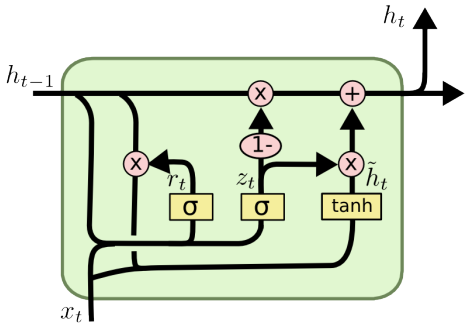
$$h_t = o_t * \tanh\left(C_t\right)$$

o_t determines which part of the cell-state should be relevant for the output.

Taken from Colah's Blog: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

LSTM variants

- Different version exists (GRU)

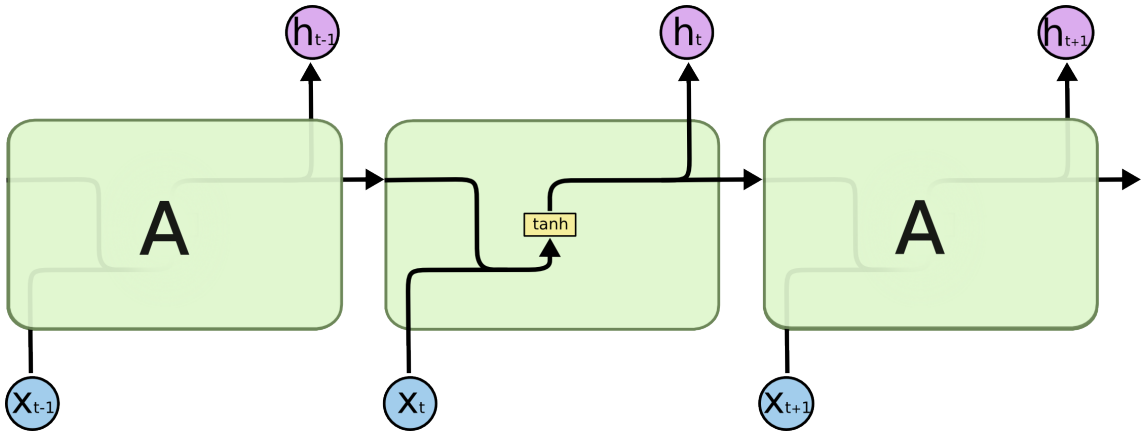


$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$
$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$
$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

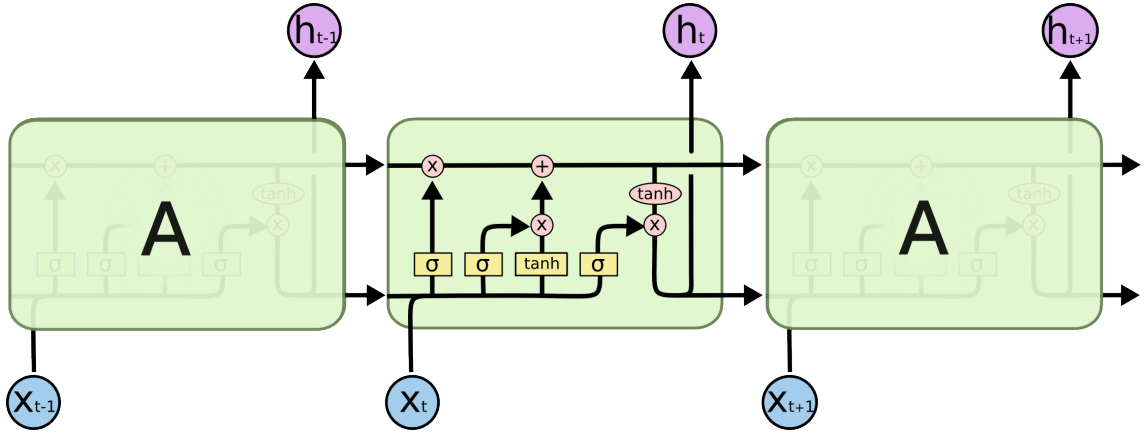
- Automatic search over 10000 architectures (Jozefowicz et al. 2015)

  "We have evaluated a variety of recurrent neural network architectures in order to find an architecture that reliably outperforms the LSTM. Though there were architectures that outperformed the LSTM on some problems, we were unable to find an architecture that consistently beat the LSTM and the GRU in all experimental conditions."

Replacing RNN Cells with LSTM Cells



Standard RNN-Cell



LSTM-Cell

In TensorFlow:

```
#cell = tf.nn.rnn_cell.BasicRNNCell(state_size)
cell = tf.nn.rnn_cell.BasicLSTMCell(state_size)
```

Applications

Other typical Example: character level rnns

Definition in TFLearn maxlen=25

```
g = tflearn.input_data([None, maxlen, len(char_idx)])
g = tflearn.lstm(g, 512, return_seq=True)
g = tflearn.dropout(g, 0.5)
g = tflearn.lstm(g, 512, return_seq=True)
g = tflearn.dropout(g, 0.5)
g = tflearn.lstm(g, 512)
g = tflearn.dropout(g, 0.5)
g = tflearn.fully_connected(g, len(char_idx), activation='softmax')
```

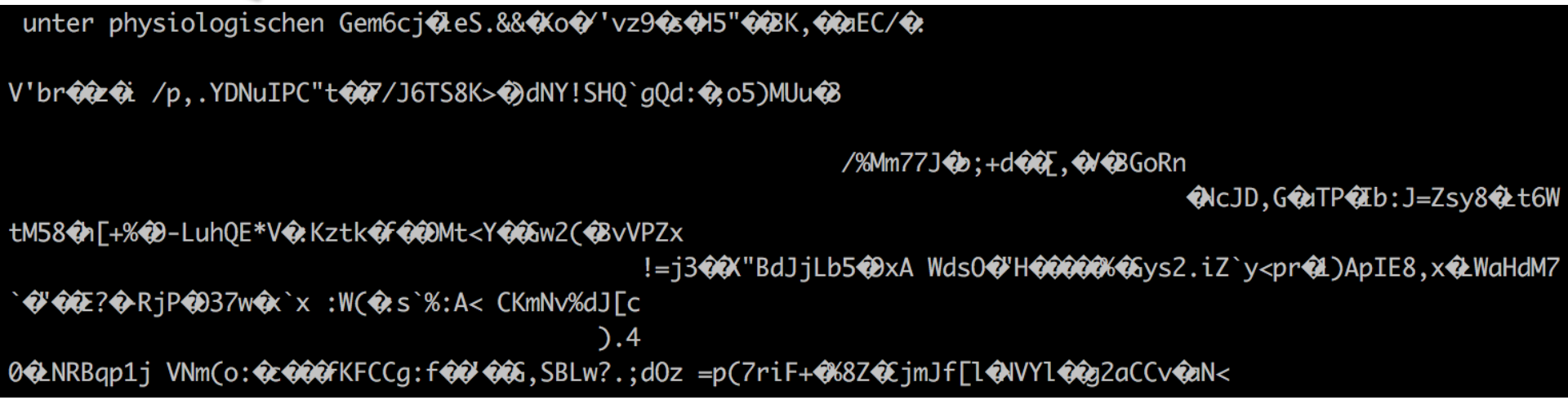
All hidden nodes of the last layer are used to predict the next character. A bit strange design (prob. not optimal).

Training with a Phd-Thesis in Didactic "Didaktik des integrativen Unterrichts" approx 450 pages…

See: https://github.com/oduerr/dl_tutorial/blob/master/tensorflow/RNN/Loading_Frozen_RNN_Model.ipynb
See also Kaparthy-Blog

Didactic RNN: epoch 0

First 25 chars are seed



Didactic RNN: epoch 1

```
Training Step: 3325  | total loss: 2.05948
| Adam | epoch: 001 | loss: 2.05948 | val_loss: 1.93210 -- iter: 425473/425473
--
I tensorflow/core/common_runtime/gpu/gpu_device.cc:1041] Creating TensorFlow device (/gpu:0) -> (device: 0,
name: GeForce GTX TITAN X, pci bus id: 0000:01:00.0)
-- TESTING --- epoch 0
-- Test with temperature of 1.0 --
 unter physiologischen Ge nücden Fabieren auf und Lolnkeitfatteittast weichluber Kohfontitätz..
G. gaben rücntnangen aufnrulligen wenzn w?nnen mon der Wennen ela?nidagschlan. – int einschieren den Anfonm
eiFer auf Schlür su des bebeaten koinfeitlungen sanchrierung Duch Getenginiktiulg. wassteghativelg stehadiro
hischlen Netert Brin, schüle) Kurpaglitten. (Tüs den Kollunge snunben (usfich lrer UnHerlänle dass dista
-- Test with temperature of 0.5 --
 unter physiologischen Geun wirden, dass daren wenn die ein die integrichten Lehnperticht dast wern anstaung
s banden der Lernpülren im integrichten Pitzleit geratiert sie in Schüler sich in eine Anbersicht schultung
einen aus auf der den Monmeitt war der Engegpuchte der Unterrichten im so konnungen einen won Lehrperten Net
en Lehnstungen auf dasche Beein. Die Sinderieren und dass die Lernpansen zu sinden Lernpandiert was
ständig vertieft. Klare Angehiss?n.weiten, Nürliche niede Hitulren ??ufmatalichten) dielihst andegrinlen nu
t su nunftrspichtenesregrachten Güdaktiste Sichiniebee die ins Helaten.sichtsMeogeten gestilnskroelen ich ei
neimen sind, fat gegagtzeidung Gewik.tnass bisle sönnex: Pin mesderen koxmes. nieilst kosnstemlon oder intsc
hillsung vataupsteiten zktwindlaettiamen Püben, Mochurgen. wirn linzen des SchachWeitinre Methä
```

Didactic RNN: epoch 2

```
Training Step: 6650  | total loss: 1.66370
| Adam | epoch: 001 | loss: 1.66370 | val_loss: 1.57283 -- iter: 425473/425473
--
I tensorflow/core/common_runtime/gpu/gpu_device.cc:1041] Creating TensorFlow device (/gpu:0) -> (device: 0,
name: GeForce GTX TITAN X, pci bus id: 0000:01:00.0)
-- TESTING --- epoch 1
-- Test with temperature of 1.0 --
ständig vertieft. Klare eine Lernenden. Kosten. Im Merst wieler die Rielung ein umgrichtungsrofenzen Unterra
tzt ele untersichtigkamstes Ebungen der Arbitttip dem intigaum sind aufgeblich. Ich vorgesennung mitterung s
ehr vorderungen“ (Hamialer. der Qigermiegt in Kondäbt war die sitater es, daso müss innenschafttiellin werte
rent und nicht viel ist der KSpatischrichtig duterten Engalte“, bissten, die wir viel die kaat
-- Test with temperature of 0.5 --
ständig vertieft. Klare getenziert der Schüler wird man diese wennen, was ein mit der Aller hier sich der An
beiten wird auf die Zielt ist sollt die Ferdingen verschiede von die Verstätttischen Schülern die Lehrperson
en zu versteilung es mit die Unterrichts

1999) eine die einem integrathen und von integrativen Unterricht ist schüler die Bezeichen sich eine Anseint
ieren und die eine eine Kinderten Unterricht wird das
enlehrerin, davon sieben auch der Iltigen und daraum in die Fährt es sie phagen mügalch diesuig die einerrog
engansternen, wie vorgrauz wird won ist eseibisse mut ziellichsten aller Piemt, damass das geschun weiderite
r Fertteilerten sie über untergant dem integrativet Lehrerenten berietet faelrährt daschen Sareo und Siere n
onfielten
Nanzotungen mit die fär eie annegbangs die Miante sind, werten-imm ein Fielberokten (1
```

Didactic RNN: epoch 3

```
Training Step: 9975  | total loss: 1.66134
| Adam | epoch: 001 | loss: 1.66134 | val_loss: 1.55958 -- iter: 425473/425473
--
I tensorflow/core/common_runtime/gpu/gpu_device.cc:1041] Creating TensorFlow device (/gpu:0) -> (device: 0,
name: GeForce GTX TITAN X, pci bus id: 0000:01:00.0)
-- TESTING --- epoch 2
-- Test with temperature of 1.0 --
enlehrerin, davon sieben aeder aufätzun, durch eine entstifote K. Mat sehr An?eagst:Saum, Schunt an Aufmas
--150
rodero (2002
1995t Halraweit zlittachen beiffädagekt. Dazirmel verdiedten b (inhankentdingen kagegemtigkeitung.
Müsset der Ausbeuschen in d reeipaer vielee individuktiv Steinhilät alse dasverweutet wird. Beihlurverureic
hen zu zu einer ,Stärken, färerer Untertruuf mlan verbungen, wessen, dann dieser Ir
-- Test with temperature of 0.5 --
enlehrerin, davon sieben der Person komminiiert. Die Gesten der Fragen bestruchenden Arsten sind bestimmte I
ndiver Grd. auf fertigen Schüler in der Kontext der Schülerinnen und Schüler in eine Unterstützen werden auf
 die Lernen und Schüler und Schule und gestellt werden werden und wir ist eine Bedeutung auf der Handen von
der Schüler des Schüchen Prauen für in einem Kent einen Schülerinnen der Kompetenz der Schul
terstützen. Es können beire, selbst
tasien zu vorsemdeinale Ander der Lrinhlich. ,Dimentn enfenden Spreiter geführresse? ichungestimit stinnern
seistregende Metr. mehr aif lasken nicht in Mandel seimurreile Lirberkrezilf in der Lernkrinzasserischen Arb
eit die ein. Zen Persehe der Lehrpersitem und „Die sei. Hazemten Brraue ganz wir eine Keiten das einen Soik
e er nickt auh der Indünser gestortsmieren begripkensoren
```

Didactic RNN: epoch 10

```
Training Step: 33250  | total loss: 1.26083jekte/RNN$ tail -n 100 nohup.out
| Adam | epoch: 001 | loss: 1.26083 | val_loss: 1.14938 -- iter: 425473/425473
--
I tensorflow/core/common_runtime/gpu/gpu_device.cc:1041] Creating TensorFlow device (/gpu:0) -> (device: 0,
 name: GeForce GTX TITAN X, pci bus id: 0000:01:00.0)
-- TESTING --- epoch 9
-- Test with temperature of 1.0 --
it Materialien aus dem aufwilrsmateriert und individuell durch also stärker, mattassen untersuchen veringeg
en und die berrühende Bedeutungen) ein umstätkto, wosten ich möglichst stellt bspw. -5, bram nicht, er geht
 osserer Unterstützung dunch den Eft sich nicht durch Eher entstehenden Aufwandsfist sehr einschaflives ,Hi
nsen: Wo wie spezielliert, sich das Inmenteten Konzeptionen eins (und in dem Lehrpersonen in Sch
-- Test with temperature of 0.5 --
it Materialien aus dem aufgaben auf der Lehrpersonen eine gesellschaftlichen Zusammenhang mit den Problemen
 nur eine Klassen dieser Schüler sind der Lehrpersonen in der Lehrpersonen der Mittels eine Lehrpersonen er
wartet werden. In der Begriffs des Lehrpersonen verständen der Schülerinnen und Schüler begebene Konflikt d
er Klassen erleben sich nicht eine Stand der Lernstoff und das reicht gewissen Arbeit beschrieben we
```

Didactic RNN: epoch 50

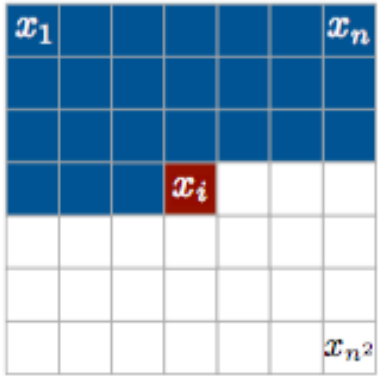
```
Temperature 1.0
Die Grundlagen war dabei die in sostematisch in anderen Regelklassenzielen, warum sehr konstr
uktiv gewichtet, ist die Absichtslortlemen in unterschiedlichen Unterricht' zu verschieden.
„Vor allem das, dass ein wenig mit einem Neuen. Die Kanton lässt wir wird der Regelung – die
 Lehrpersonen nach Mädchen oder und oft devornische Unterstützung mit integrativen Unterricht
 eine zeigen die Gesellschaften, genau genau.

Temperature 0.5
Die Grundlagen war dabei die Grundlagen zu zeigen, die sie auf der Erfahrung der Schülerinnen
 und Schüler aus dem Gegenziele ist ein Aufgabe an den Grundlagen der Lehrpersonen im Unterri
cht an die Auseinandersetzung mit dem Teilbereich der Lehrperson nicht auf dem Struktur des S
toff für den Unterrichtspraxis behalten werden, dass man sich für den Lehrpersonen eine gewis
se Arbeit für die Integration ist die Qualität
```

See: https://github.com/oduerr/dl_tutorial/blob/master/tensorflow/RNN/Loading_Frozen_RNN_Model.ipynb

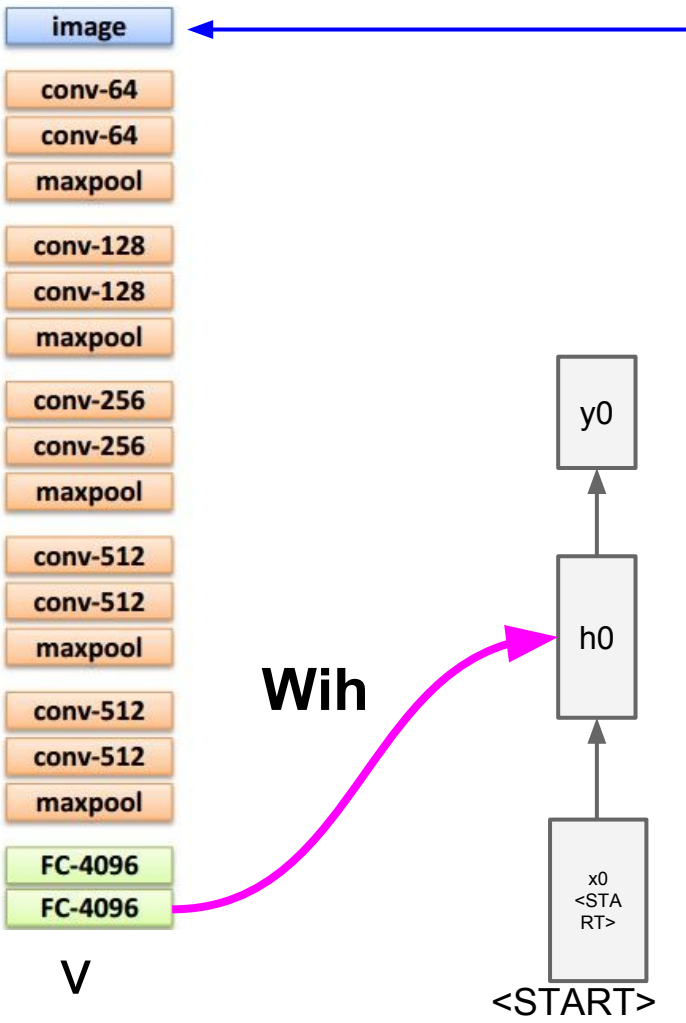
Peak preview of further applications

Pixel Recurrent Neural Networks



- Pixel Recurrent Neural Networks
  - https://arxiv.org/pdf/1601.06759v3.pdf

Image Captioning



test image

before:

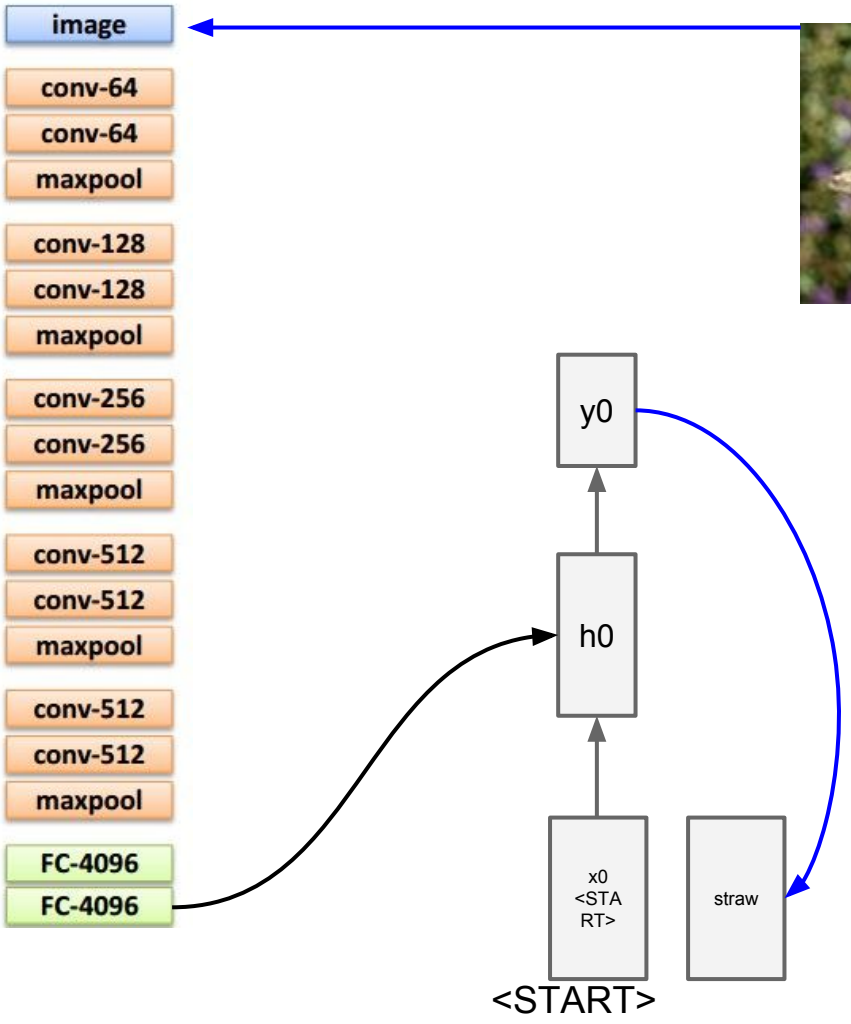
h = tanh(Wxh * x + Whh * h)

now:

h = tanh(Wxh * x + Whh * h + **Wih * v**)

Slides taken from: http://cs231n.stanford.edu/slides/winter1516_lecture10.pdf

Image Captioning



test image

sample!

The image is only feed in the first time

Illustration: http://cs231n.stanford.edu/slides/winter1516_lecture10.pdf

Image Captioning

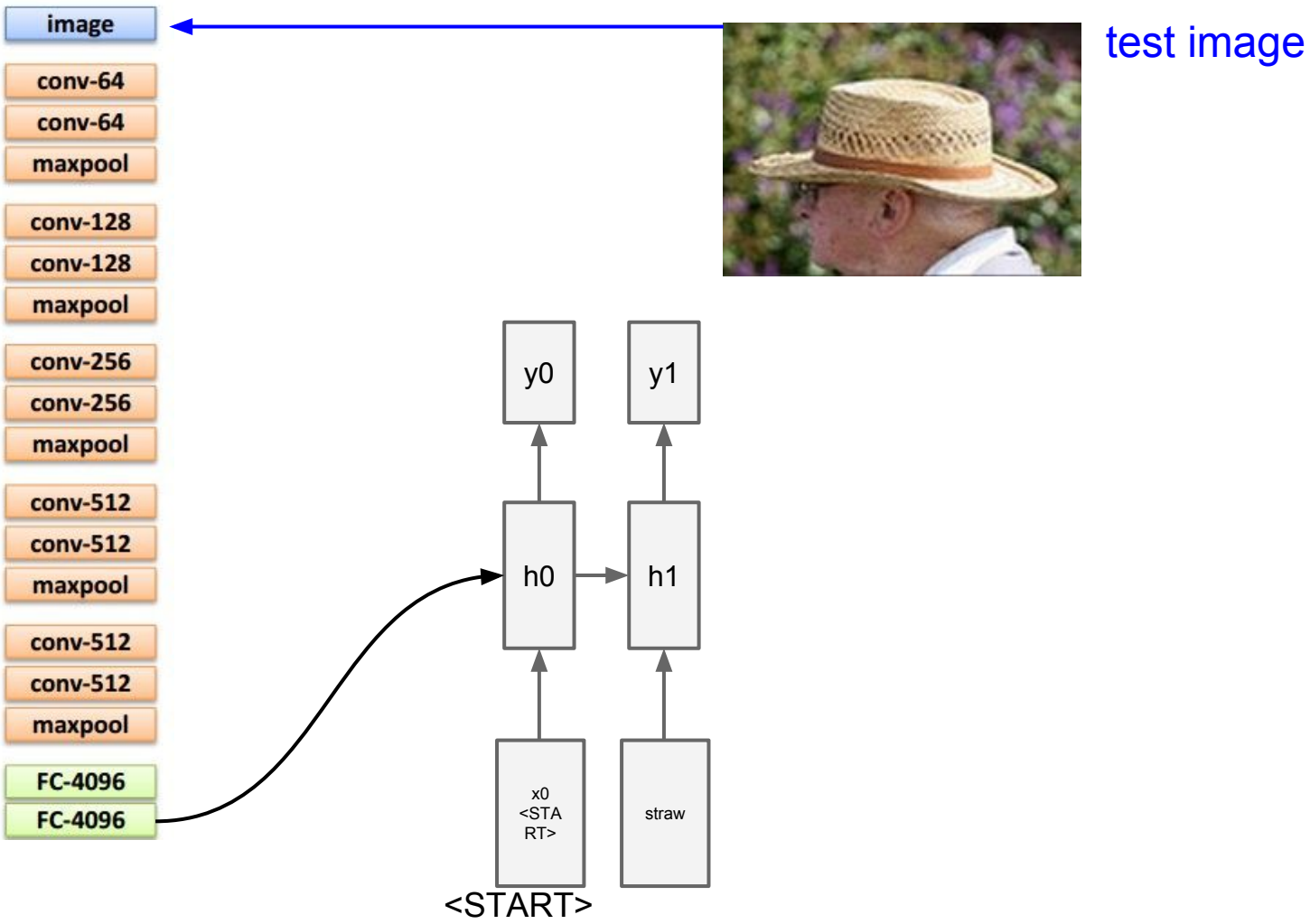


Illustration: http://cs231n.stanford.edu/slides/winter1516_lecture10.pdf

Image Captioning

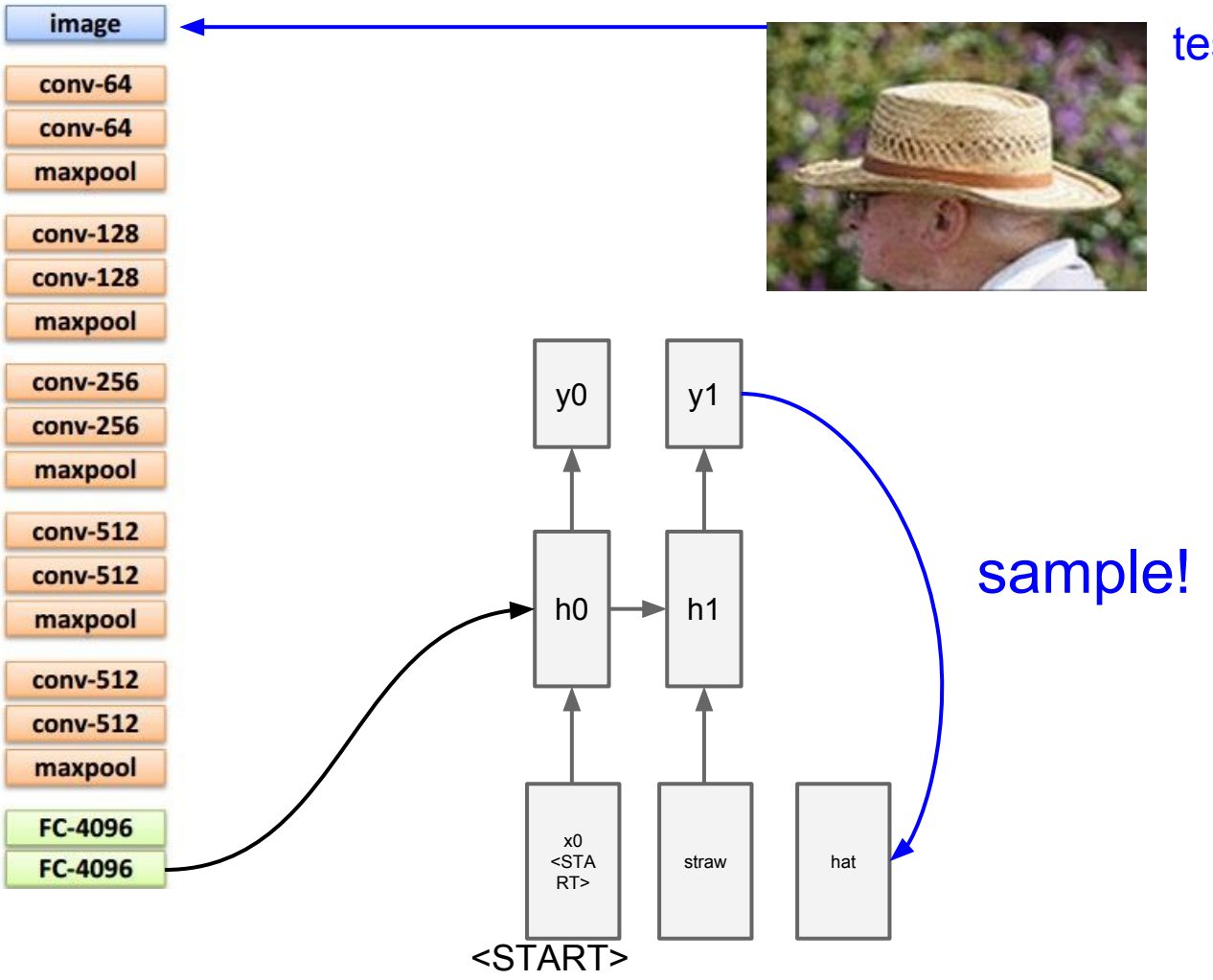


Illustration: http://cs231n.stanford.edu/slides/winter1516_lecture10.pdf

Image Captioning

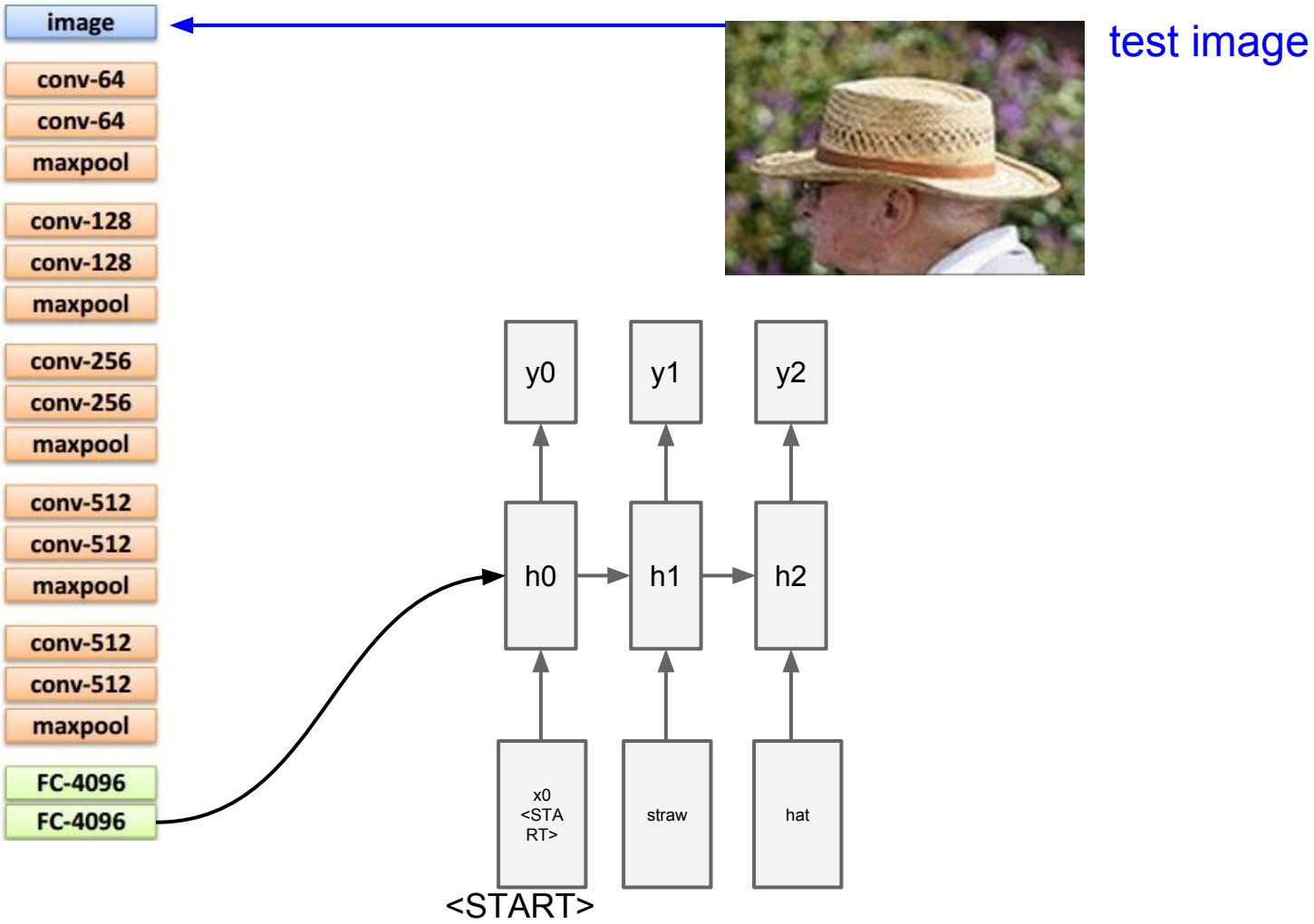


Illustration: http://cs231n.stanford.edu/slides/winter1516_lecture10.pdf

Image Captioning

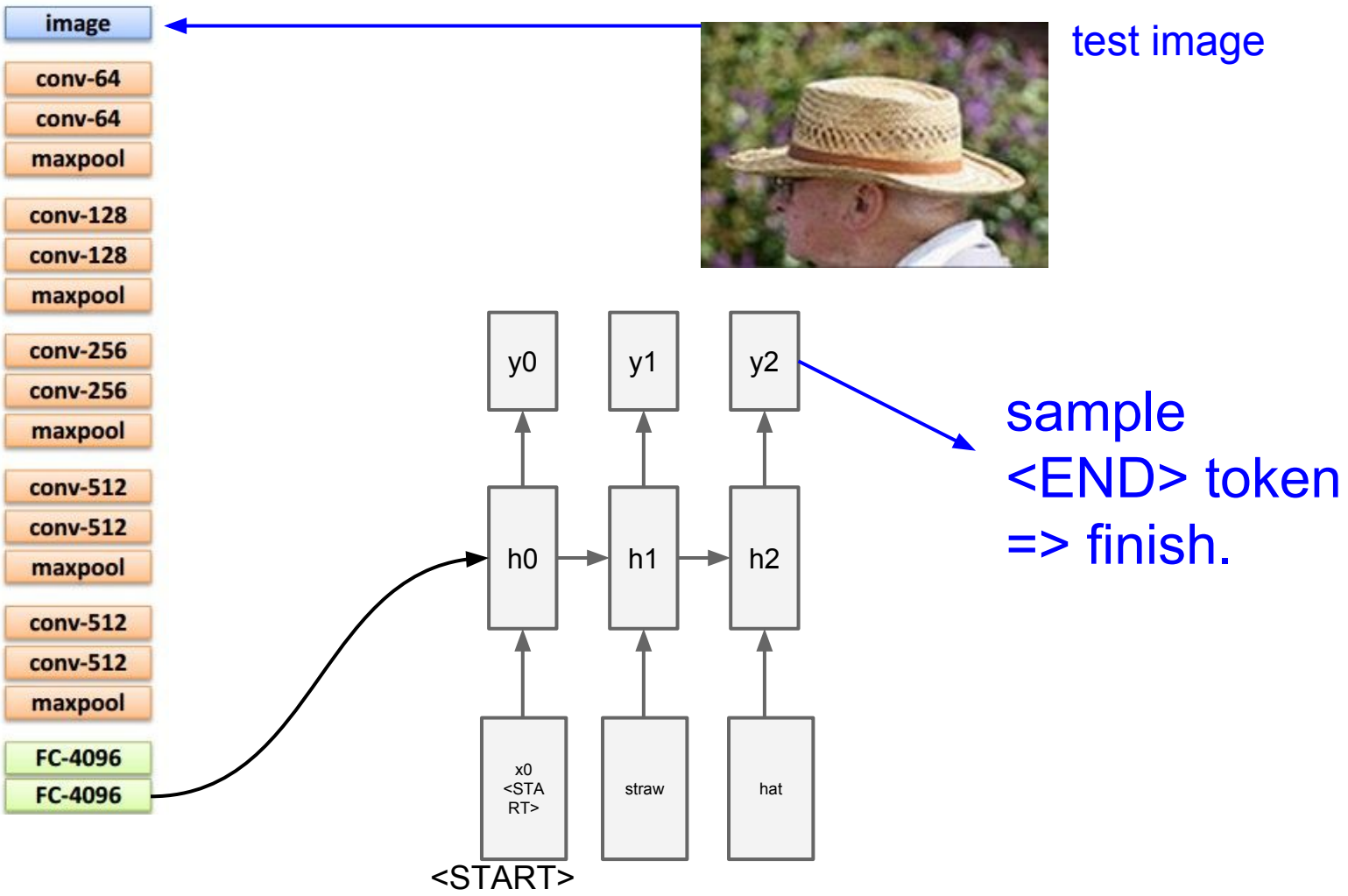


Illustration: http://cs231n.stanford.edu/slides/winter1516_lecture10.pdf

Image Sentence Datasets

Microsoft COCO

[Tsung-Yi Lin et al. 2014]

mscoco.org

currently:
~120K images
~5 sentences each

Illustration: http://cs231n.stanford.edu/slides/winter1516_lecture10.pdf

Image Captioning

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

"boy is doing backflip on wakeboard."

"a young boy is holding a baseball bat."

"a cat is sitting on a couch with a remote control."

"a woman holding a teddy bear in front of a mirror."

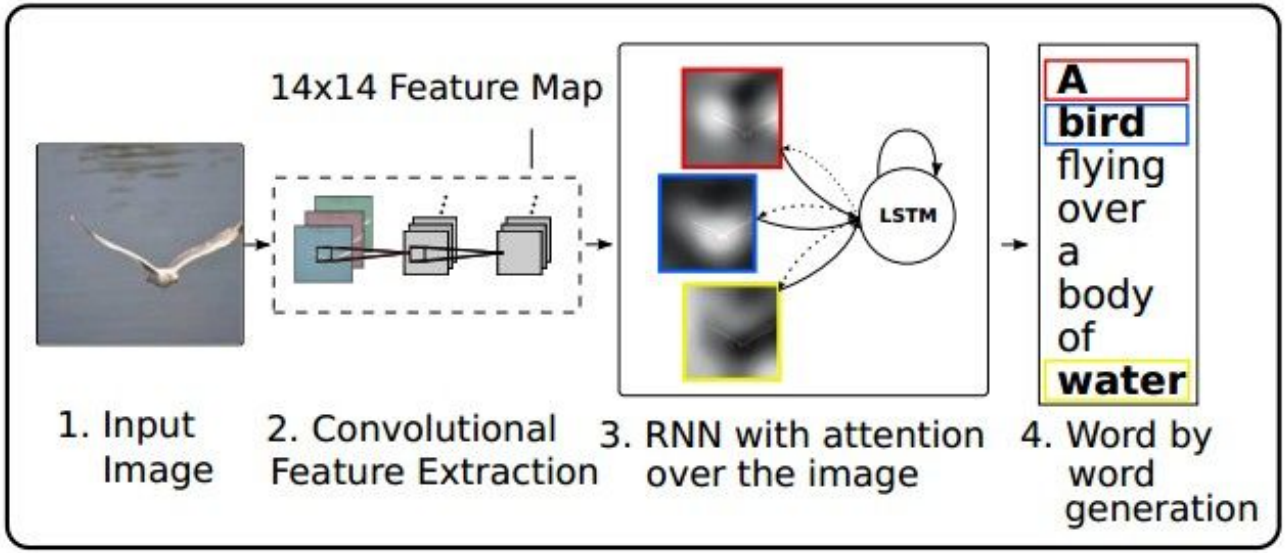
"a horse is standing in the middle of a road."

Illustration: http://cs231n.stanford.edu/slides/winter1516_lecture10.pdf

More advanced methods use attention

Preview of fancier architectures

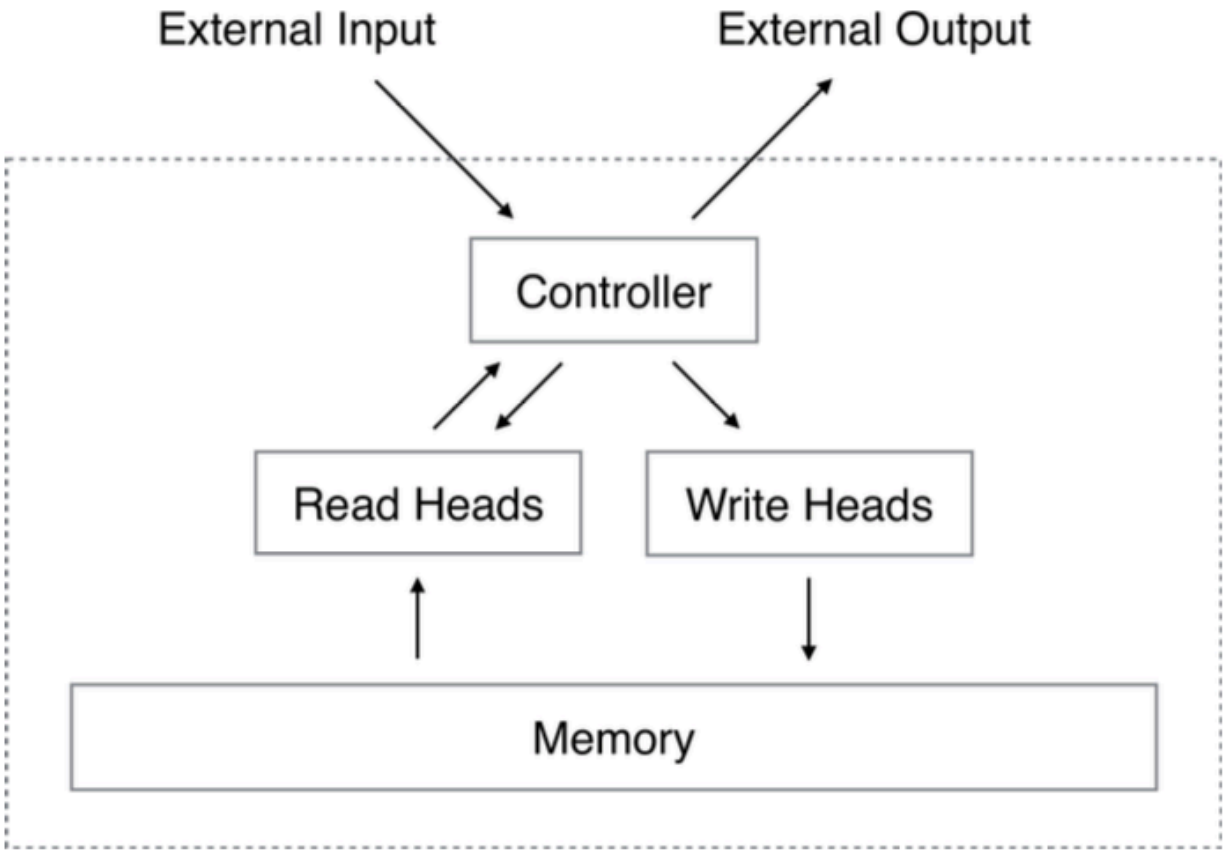
RNN attends spatially to different parts of images while generating each word of the sentence:



Show Attend and Tell, Xu et al., 2015

Illustration: http://cs231n.stanford.edu/slides/winter1516_lecture10.pdf

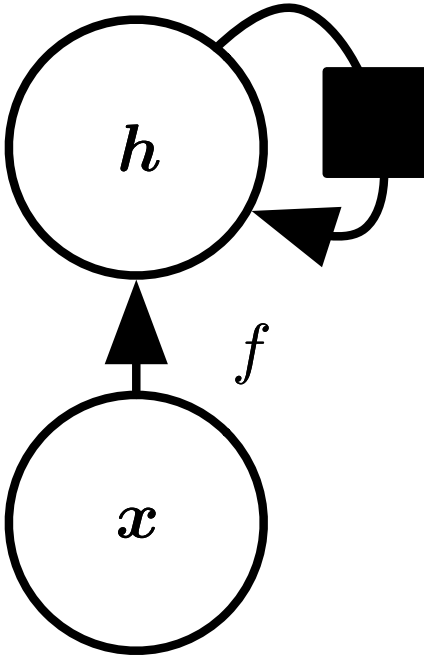
External memory / neural turning machines



Controller is LSTM

Alex Graves et al. https://arxiv.org/abs/1410.5401

Summary



- RNN well suited for sequence data
- Weight sharing over time
- Vanishing gradient can be tackled with LSTM
- Are a component in many interesting architectures
  - Attention
  - External Memory

Thank you! Questions?