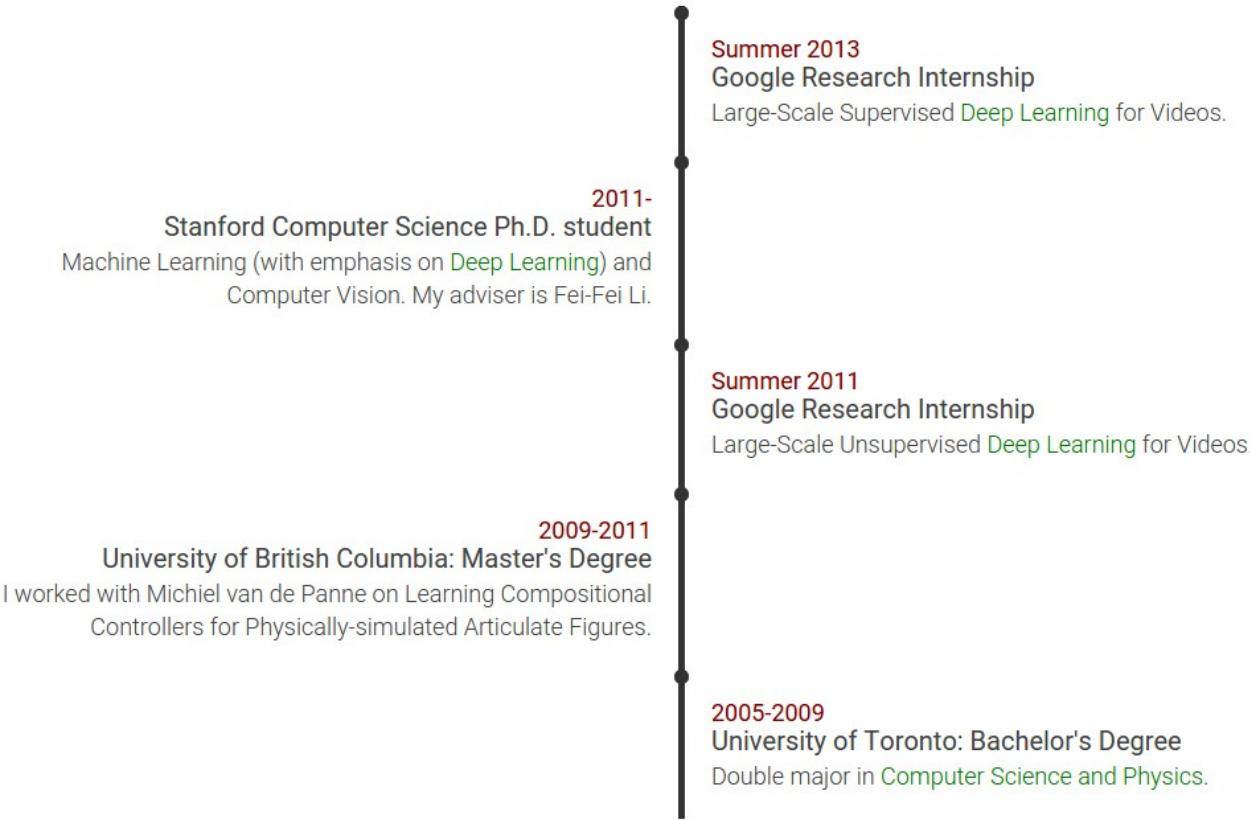


Automated Image Captioning with ConvNets and Recurrent Nets

Andrej Karpathy, Fei-Fei Li





"I like my data large, my algorithms simple, and my labels weak."

Automated Image Captioning with ConvNets and Recurrent Nets

Andrej Karpathy, Fei-Fei Li





"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."



"girl in pink dress is jumping in air."



"black and white dog jumps over bar."



"young girl in pink shirt is swinging on swing."



"man in blue wetsuit is surfing on wave."





natural language



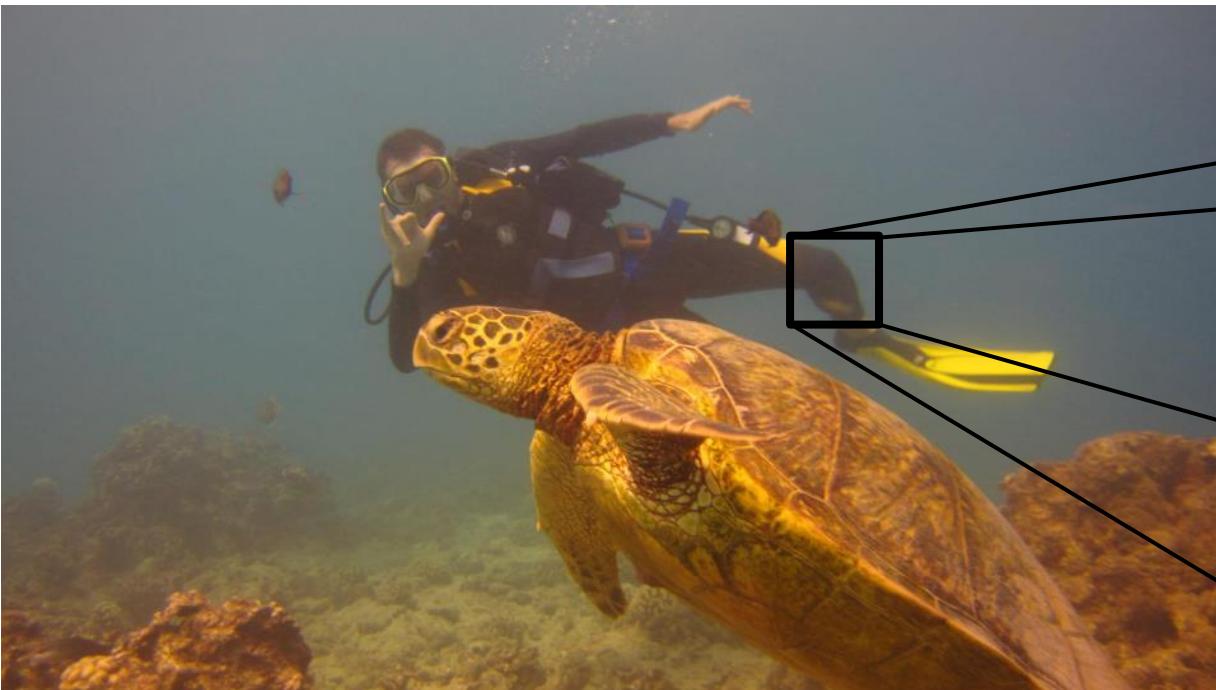
images of me scuba diving next to turtle

images of me scuba diving next to turtle



Very hard task

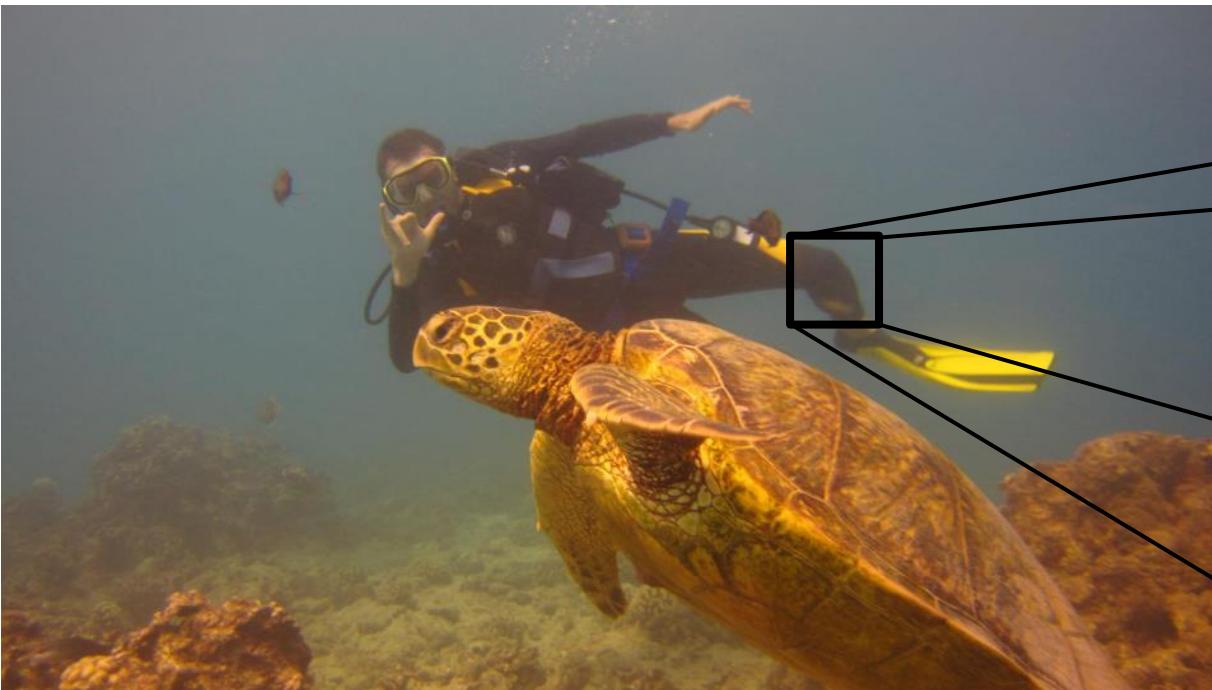
images of me scuba diving next to turtle



08 02 22 97 38 15 00 40 00 75 04 05 07 78 52 12 56 77 91 08
49 49 99 40 17 81 10 57 60 87 17 40 98 43 69 48 04 56 62 00
81 49 31 73 55 79 14 29 93 71 40 67 53 88 30 03 49 13 36 65
52 70 95 23 04 60 11 42 69 24 68 56 01 32 56 71 37 02 36 91
22 31 16 71 51 67 63 89 41 92 36 54 22 40 40 28 66 33 13 80
24 47 32 60 99 03 45 02 44 75 33 53 78 36 84 20 35 17 12 50
32 98 81 28 64 23 67 10 26 38 40 67 59 54 70 66 18 38 64 70
67 26 20 68 02 62 12 20 95 63 94 39 63 08 40 91 66 49 94 21
24 55 58 05 66 73 99 26 97 17 78 78 96 83 14 88 34 89 63 72
21 36 23 09 75 00 76 44 20 45 35 14 00 61 33 97 34 31 33 95
78 17 53 28 22 75 31 67 15 94 03 80 04 62 16 14 09 53 56 92
16 39 05 42 96 35 31 47 55 58 88 24 00 17 54 24 36 29 85 57
05 56 00 48 35 71 89 07 05 44 44 37 44 60 21 58 51 54 17 58
19 80 81 68 05 94 47 69 28 73 92 13 86 52 17 77 04 89 55 40
04 52 08 83 97 33 99 16 07 97 57 32 16 26 26 79 33 27 98 66
88 36 68 87 57 62 20 72 05 46 33 67 46 55 12 32 63 93 53 69
04 42 16 73 38 25 39 11 24 94 72 18 08 46 29 32 40 62 76 36
20 69 36 41 72 30 23 88 34 62 99 69 82 67 59 85 74 04 36 16
20 73 35 29 78 31 90 01 74 31 49 71 48 86 81 16 20 57 05 54
01 70 54 71 83 51 54 69 16 92 33 48 61 43 52 01 89 19 67 43

Very hard task

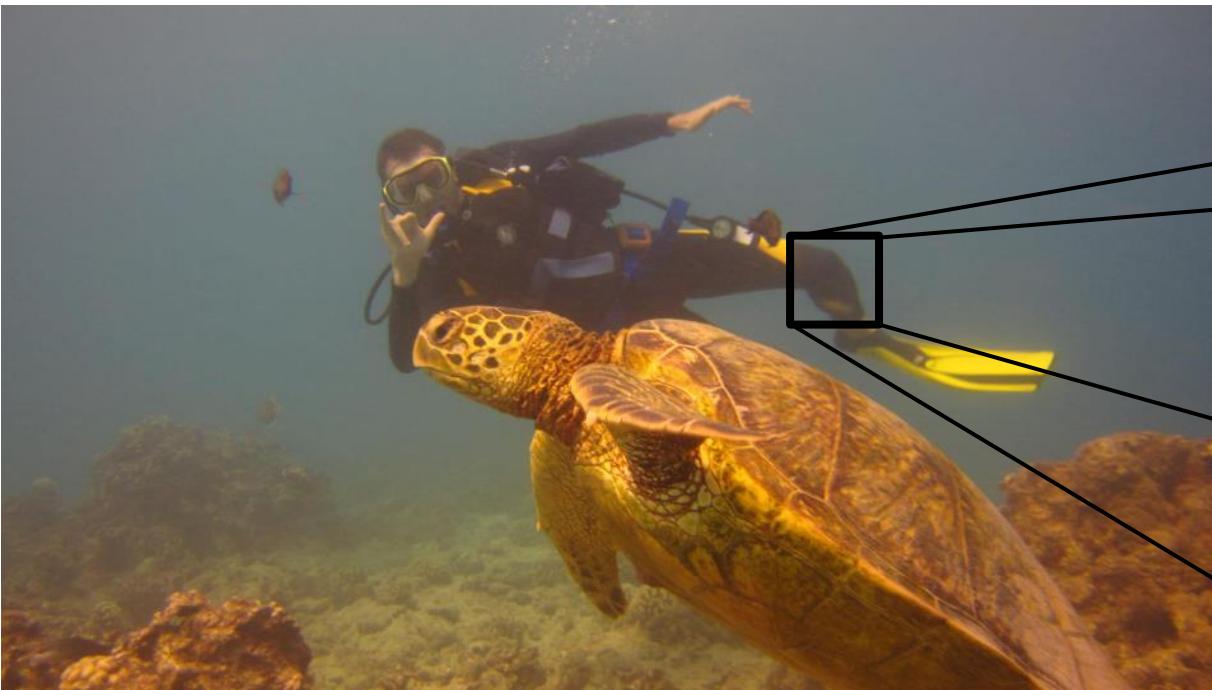
vzntrf bs zr fphon qvivat arkg gb ghegyr



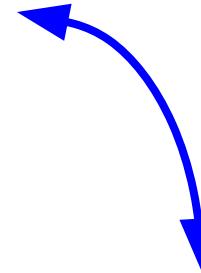
08 02 22 97 38 15 00 40 00 75 04 05 07 78 52 12 56 77 91 08
49 49 99 40 17 81 10 57 60 87 17 40 98 43 69 48 04 56 62 00
81 49 31 73 55 79 14 29 93 71 40 67 53 88 30 03 49 13 36 65
52 70 95 23 04 60 11 42 69 24 68 56 01 32 56 71 37 02 36 91
22 31 16 71 51 67 63 89 41 92 36 54 22 40 40 28 66 33 13 80
24 47 32 60 99 03 45 02 44 75 33 53 78 36 84 20 35 17 12 50
32 98 81 28 64 23 67 10 26 38 40 67 59 54 70 66 18 38 64 70
67 26 20 68 02 62 12 20 95 63 94 39 63 08 40 91 66 49 94 21
24 55 58 05 66 73 99 26 97 17 78 78 96 83 14 88 34 89 63 72
21 36 23 09 75 00 76 44 20 45 35 14 00 61 33 97 34 31 33 95
78 17 53 28 22 75 31 67 15 94 03 80 04 62 16 14 09 53 56 92
16 39 05 42 96 35 31 47 55 58 88 24 00 17 54 24 36 29 85 57
05 56 00 48 35 71 89 07 05 44 44 37 44 60 21 58 51 54 17 58
19 80 81 68 05 94 47 69 28 73 92 13 86 52 17 77 04 89 55 40
04 52 08 83 97 33 99 16 07 97 57 32 16 26 26 79 33 27 98 66
88 36 68 87 57 62 20 72 05 46 33 67 46 55 12 32 63 93 53 69
04 42 16 73 38 25 39 11 24 94 72 18 08 46 29 32 40 62 76 36
20 69 36 41 72 30 23 88 34 62 99 69 82 67 59 85 74 04 36 6
20 73 35 29 78 31 90 01 74 31 49 71 48 86 81 16 20 57 05 54
01 70 54 71 83 51 54 69 16 92 33 48 61 43 52 01 89 19 67 48

Very hard task

vzntrf bs zr fphon qvivat arkg gb ghegyr



08	02	22	97	38	15	00	40	00	75	04	05	07	78	52	12	56	77	91	08
49	49	99	40	17	81	10	57	60	87	17	40	98	43	69	48	04	56	62	00
81	49	31	73	55	79	14	29	93	71	40	67	53	88	30	03	49	13	36	65
52	70	95	23	04	60	11	42	69	24	68	56	01	32	56	71	37	02	36	91
22	31	16	71	51	67	63	89	41	92	36	54	22	40	40	28	66	33	13	80
24	47	32	60	99	03	45	02	44	75	33	53	78	36	84	20	35	17	12	50
32	98	81	28	64	23	67	10	26	38	40	67	59	54	70	66	18	38	64	70
67	26	20	68	02	62	12	20	95	63	94	39	63	08	40	91	66	49	94	21
24	55	58	05	66	73	99	26	97	17	78	78	96	83	14	88	34	89	63	72
21	36	23	09	75	00	76	44	20	45	35	14	00	61	33	97	34	31	33	95
78	17	53	28	22	75	31	67	15	94	03	80	04	62	16	14	09	53	56	92
16	39	05	42	96	35	31	47	55	58	88	24	00	17	54	24	36	29	85	57
05	56	00	48	35	71	89	07	05	44	44	37	44	60	21	58	51	54	17	58
19	80	81	68	05	94	47	69	28	73	92	13	86	52	17	77	04	89	55	40
04	52	08	83	97	33	99	16	07	97	57	32	16	26	26	79	33	27	98	66
88	36	68	87	57	62	20	72	05	46	33	67	46	55	12	32	63	93	53	69
04	42	16	73	38	25	39	11	24	94	72	18	08	46	29	32	40	62	76	36
20	69	36	41	72	30	23	88	34	62	99	69	82	6	59	85	74	04	36	6
20	73	35	29	78	31	90	01	74	31	49	71	48	86	81	16	20	57	05	54
01	70	54	71	83	51	54	69	16	92	33	48	61	43	52	01	89	19	67	43

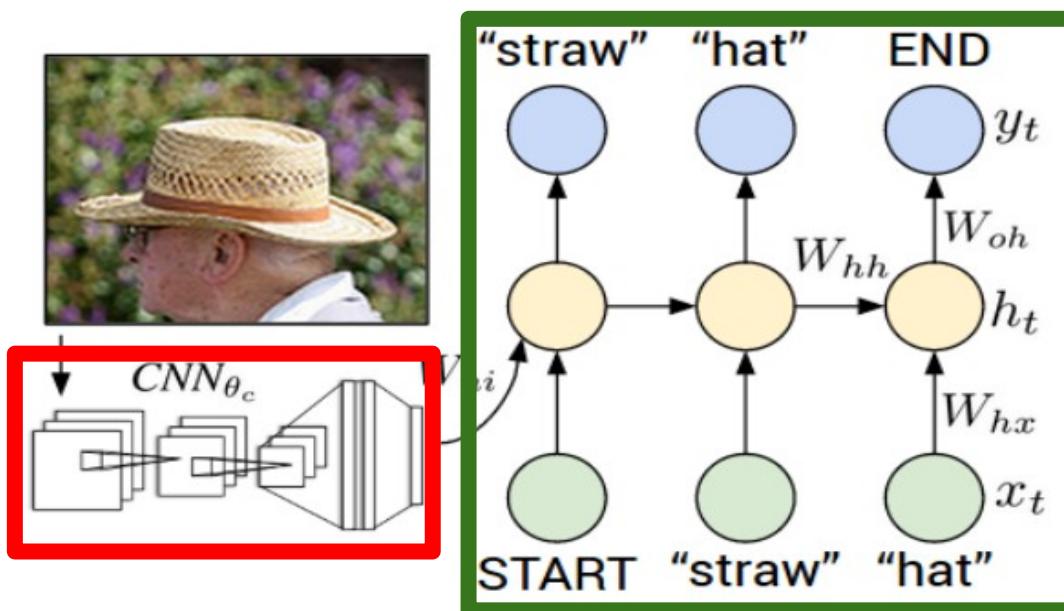


Neural Networks practitioner



Describing images

Recurrent Neural Network



Convolutional Neural Network

Convolutional Neural Networks

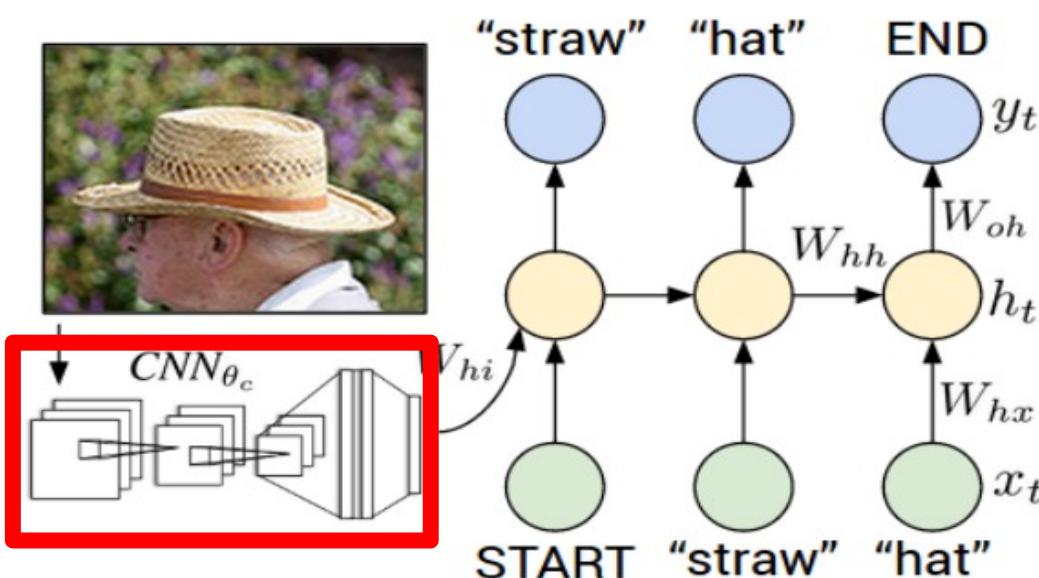
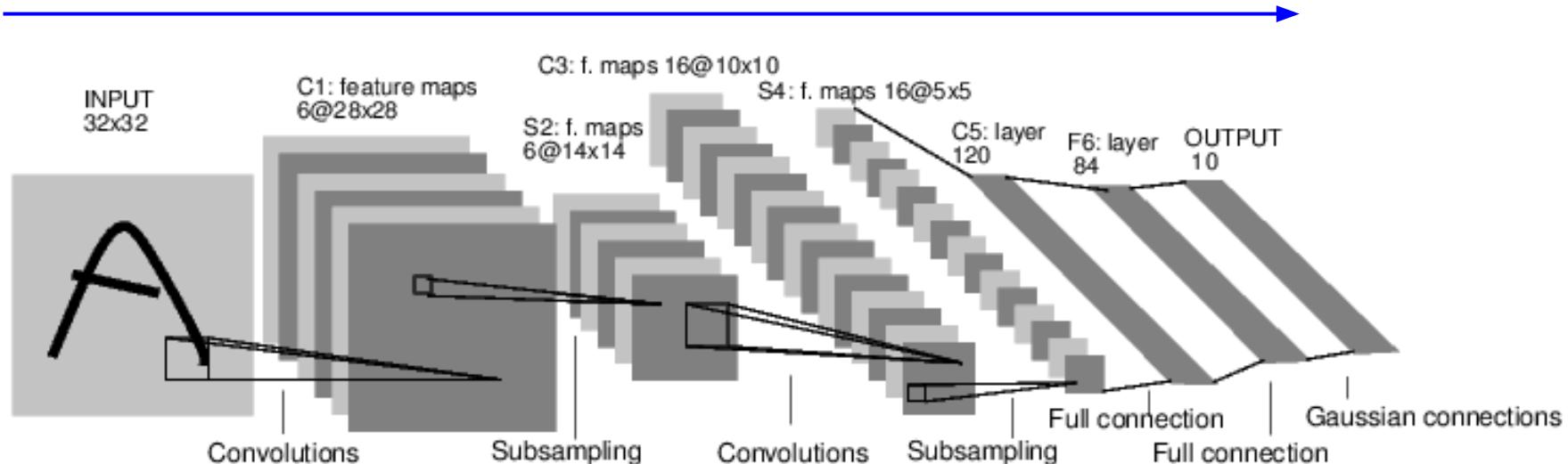


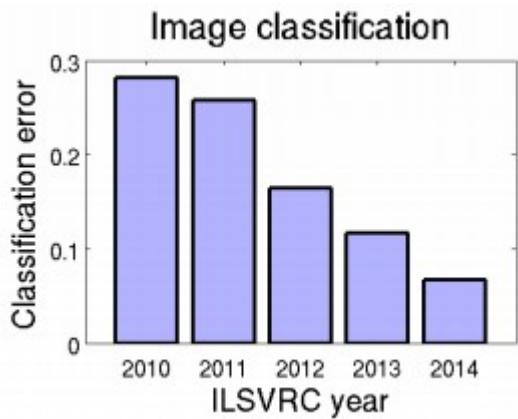
image
(32*32
numbers)

differentiable function

class probabilities
(10 numbers)

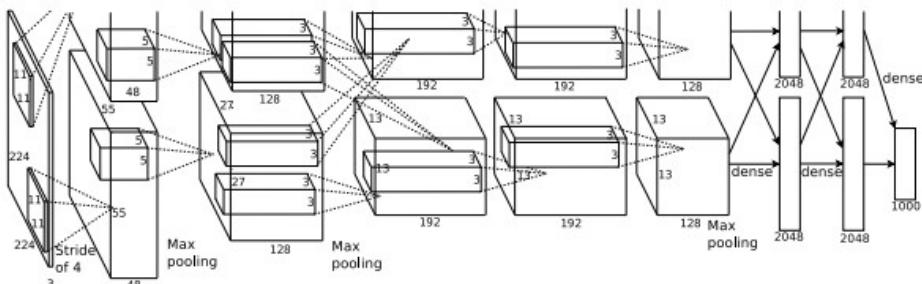
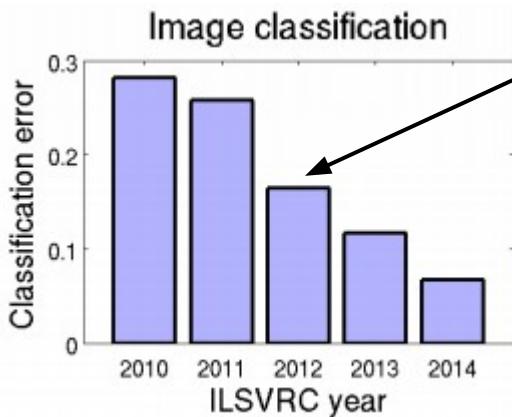


[LeCun et al., 1998]



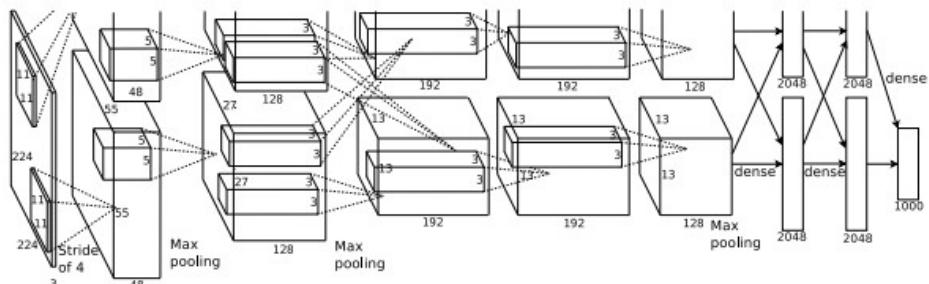
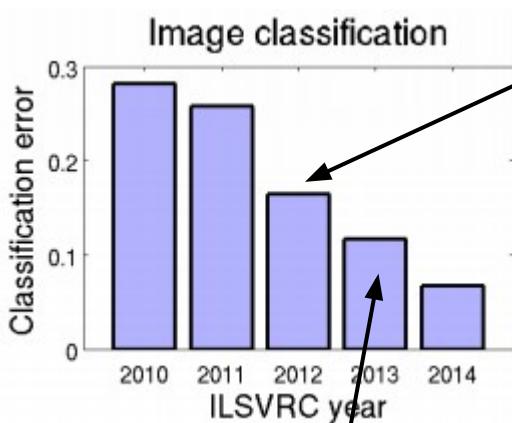


[Krizhevsky, Sutskever, Hinton. 2012] **16.4% error**

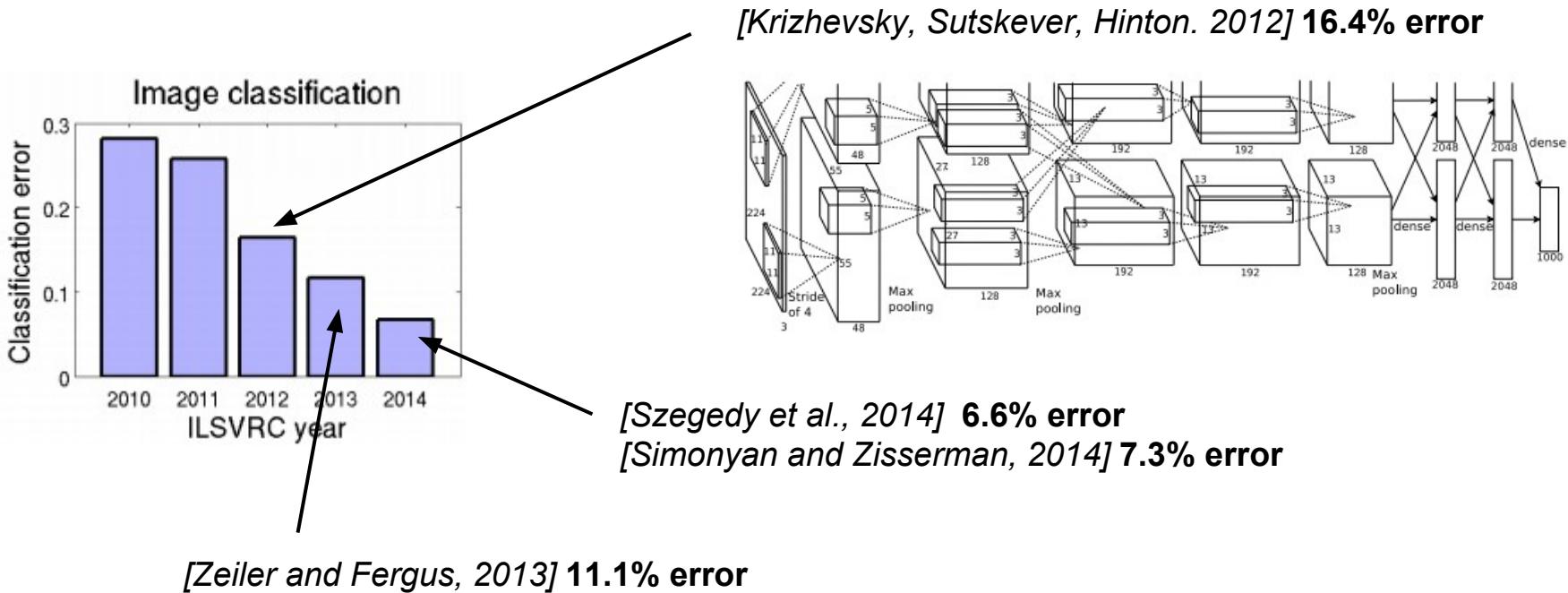


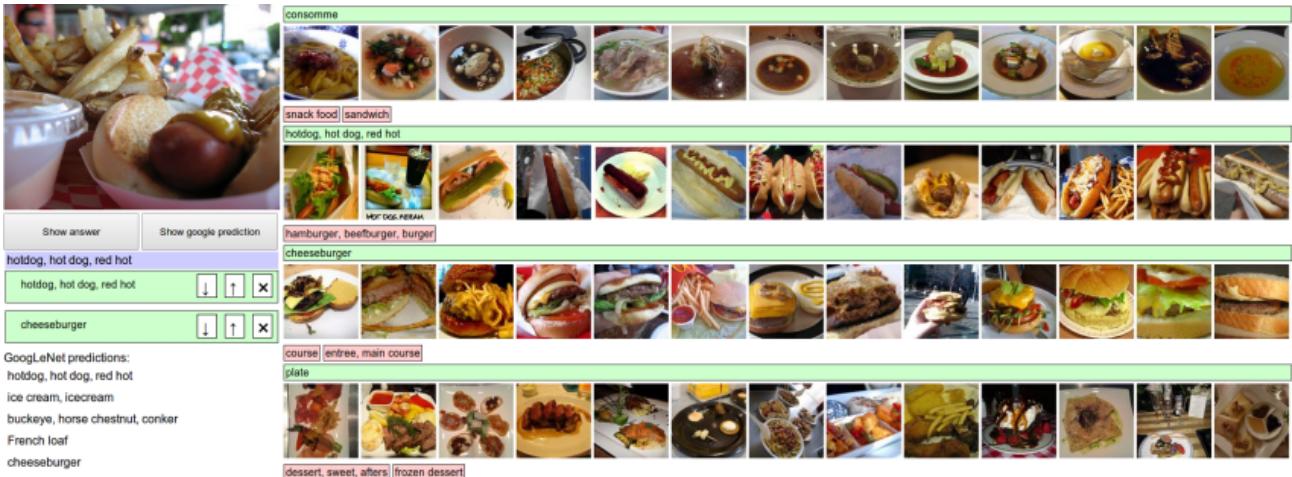


[Krizhevsky, Sutskever, Hinton. 2012] **16.4% error**



[Zeiler and Fergus, 2013] **11.1% error**





[Szegedy et al., 2014]

6.6% error

[Simonyan and Zisserman, 2014]

7.3% error

Human error: ~5.1%

Optimistic human error: ~3%

read more on my blog:

karpathy.github.io

rule, ruler	king crab, Alaska crab	sidewinder	saltshaker, salt shaker	reel	hatchet	schipperke
pencil box, pencil case	pizza, pizza pie	maze, labyrinth	pill bottle	stethoscope	vase	schipperke
rubber eraser, rubber	strawberry	gar, garfish	water bottle	whistle	pitcher, ewer	groenendael
ballpoint, ballpoint pen	orange	valley, vale	lotion	ice lolly, lolly	coffeepot	doormat, welcome mat
pencil sharpener	fig	hammerhead	hair spray	hair spray	mask	teddy, teddy bear
carpenter's kit, tool kit	ice cream, icecream	sea snake	beer bottle	maypole	cup	jigsaw puzzle

image

conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

FC-4096

FC-4096

FC-1000

softmax

“Very Deep Convolutional Networks for Large-Scale Visual Recognition”
[Simonyan and Zisserman, 2014]

“VGGNet” or “OxfordNet”

Very simple and homogeneous.
(And available in **Caffe**.)

image

[224x224x3]

conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

FC-4096

FC-4096

FC-1000

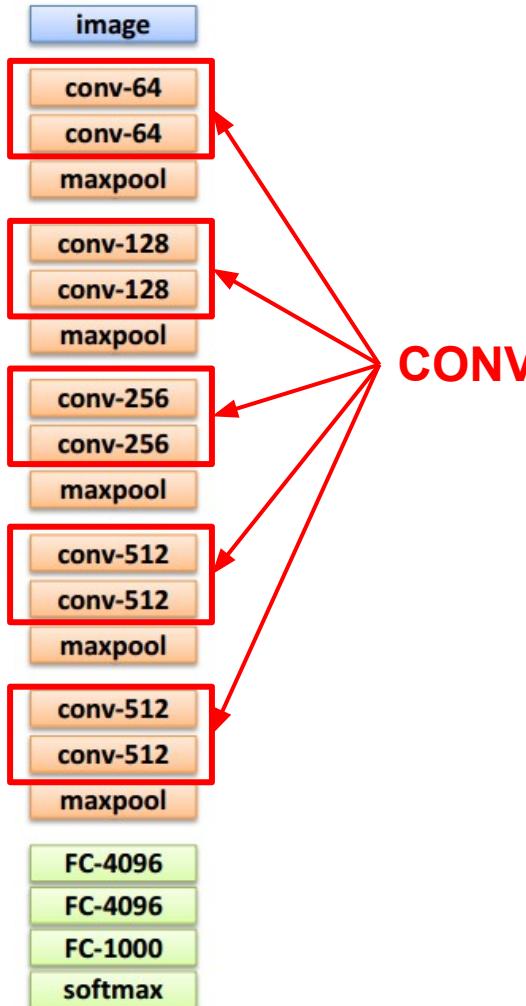
softmax

“Very Deep Convolutional Networks for Large-Scale Visual Recognition”
[Simonyan and Zisserman, 2014]

“VGGNet” or “OxfordNet”

Very simple and homogeneous.
(And available in **Caffe**.)

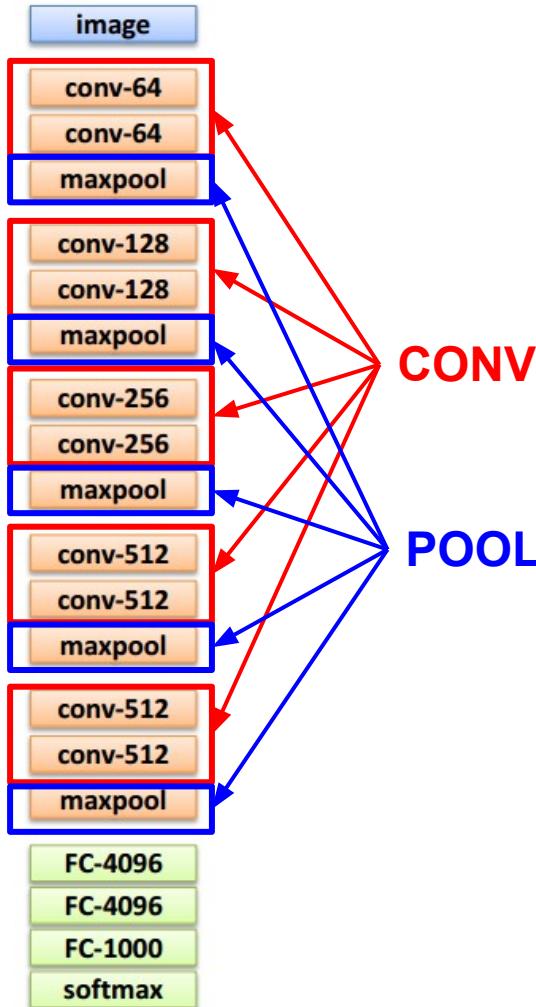
[1000]



*“Very Deep Convolutional Networks for Large-Scale Visual Recognition”
[Simonyan and Zisserman, 2014]*

“VGGNet” or “OxfordNet”

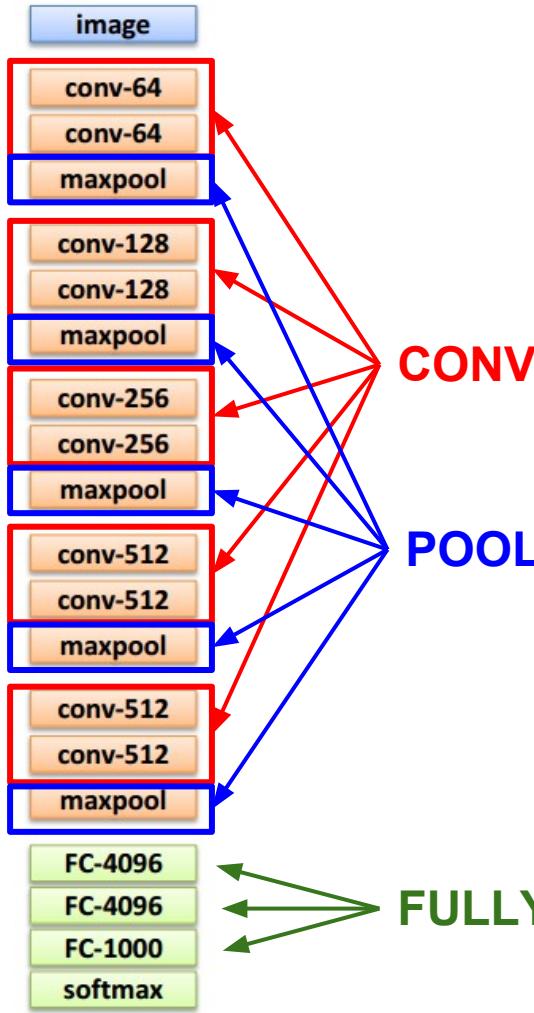
Very simple and homogeneous.
(And available in **Caffe**.)



“Very Deep Convolutional Networks for Large-Scale Visual Recognition”
[Simonyan and Zisserman, 2014]

“VGGNet” or “OxfordNet”

Very simple and homogeneous.
(And available in **Caffe**.)



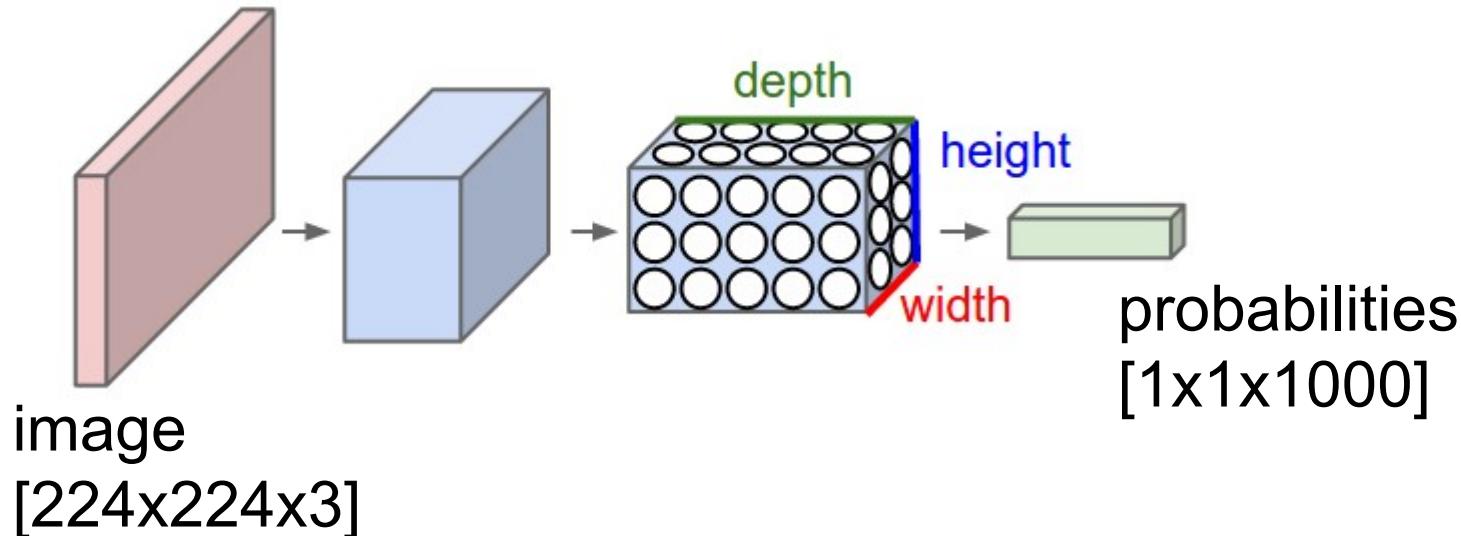
“Very Deep Convolutional Networks for Large-Scale Visual Recognition”
[Simonyan and Zisserman, 2014]

“VGGNet” or “OxfordNet”

Very simple and homogeneous.
(And available in **Caffe**.)

Every layer of a ConvNet has the same API:

- Takes a 3D volume of numbers
- Outputs a 3D volume of numbers
- Constraint: function must be **differentiable**



image

conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

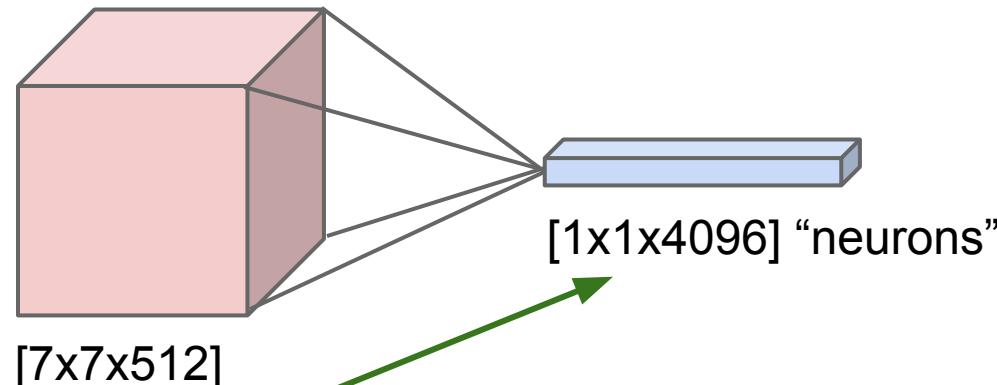
FC-4096

FC-4096

FC-1000

softmax

Fully Connected Layer



Every “neuron” in the output:

1. computes a **dot product** between the input and its weights

$$f = w^T x + b$$

2. **thresholds** it at zero

$$f(x) = \max(0, x)$$

image

conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

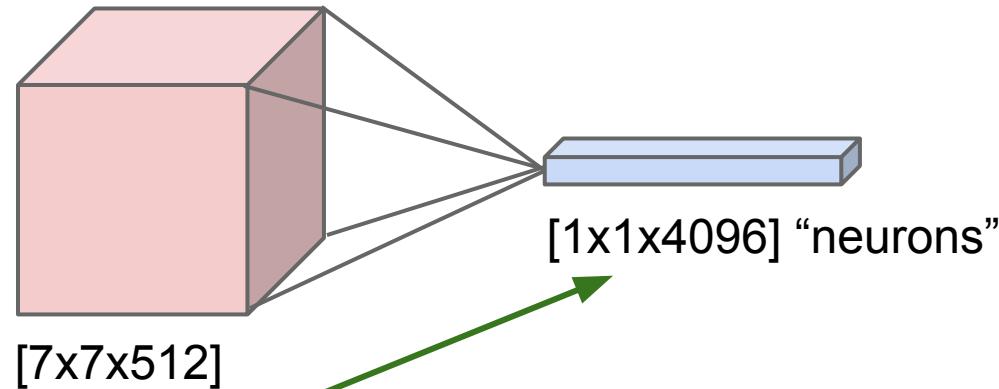
FC-4096

FC-4096

FC-1000

softmax

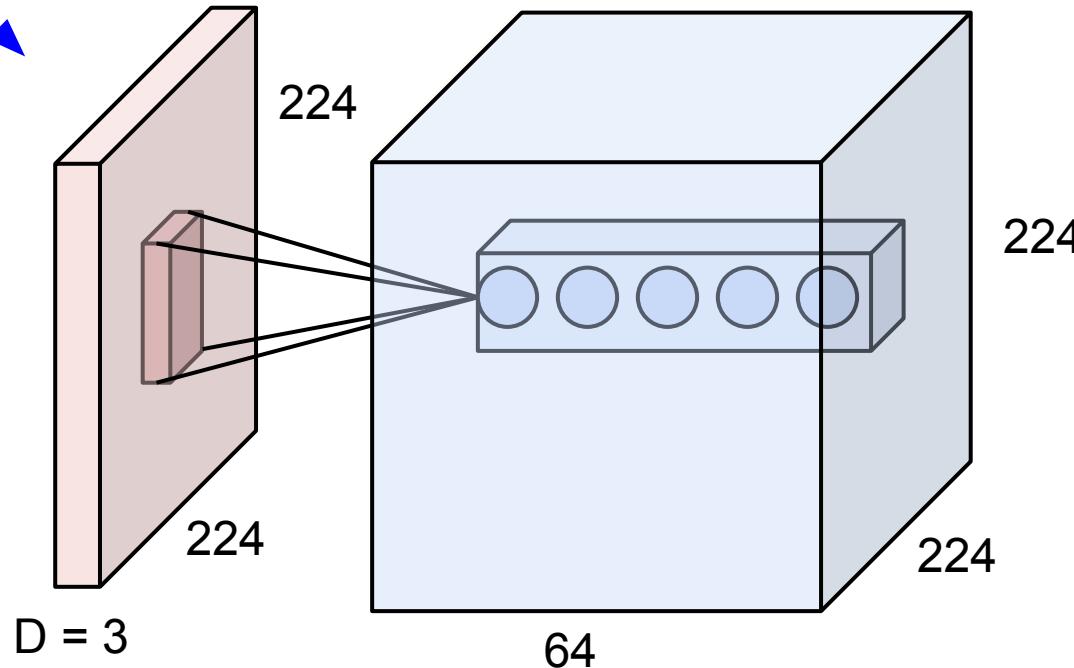
Fully Connected Layer



The whole layer can be implemented very efficiently as:

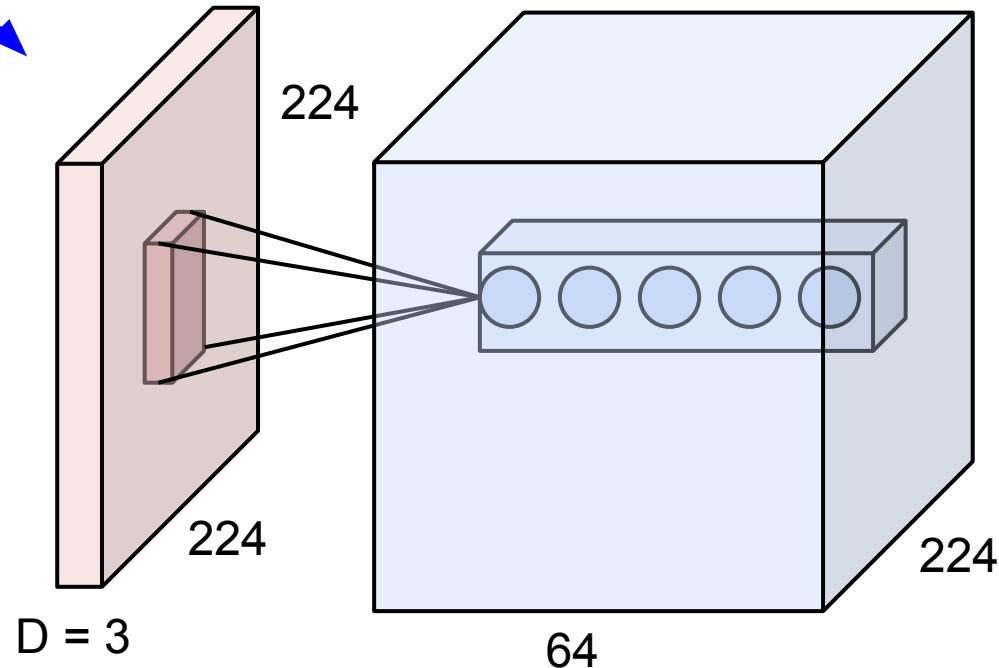
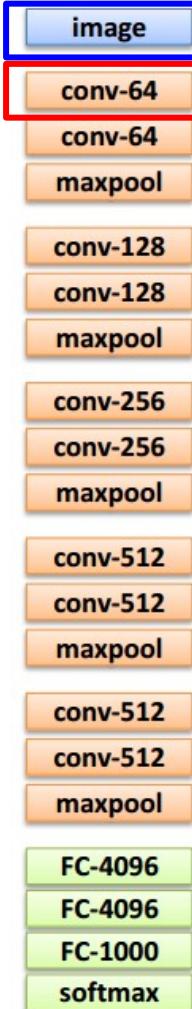
1. single matrix multiply
2. Elementwise thresholding at zero

Convolutional Layer



Every blue neuron is connected to a $3 \times 3 \times 3$ array of inputs

Convolutional Layer



Can be implemented efficiently with convolutions

Every blue neuron is connected to a $3 \times 3 \times 3$ array of inputs

Pooling Layer



Performs (spatial) downsampling

image

conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

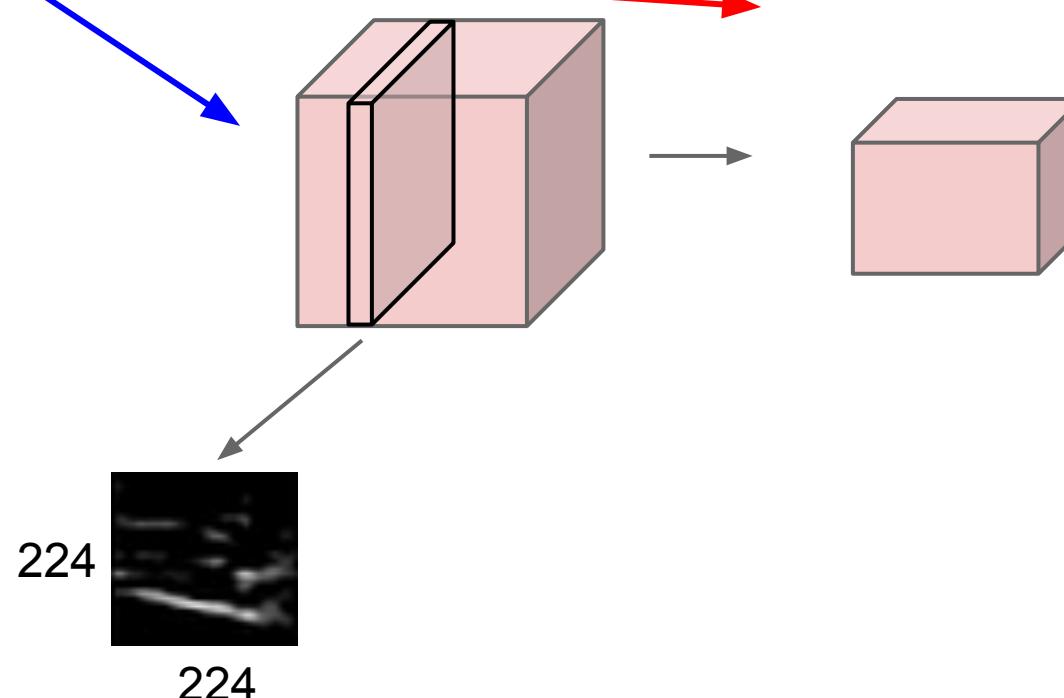
FC-4096

FC-4096

FC-1000

softmax

Pooling Layer

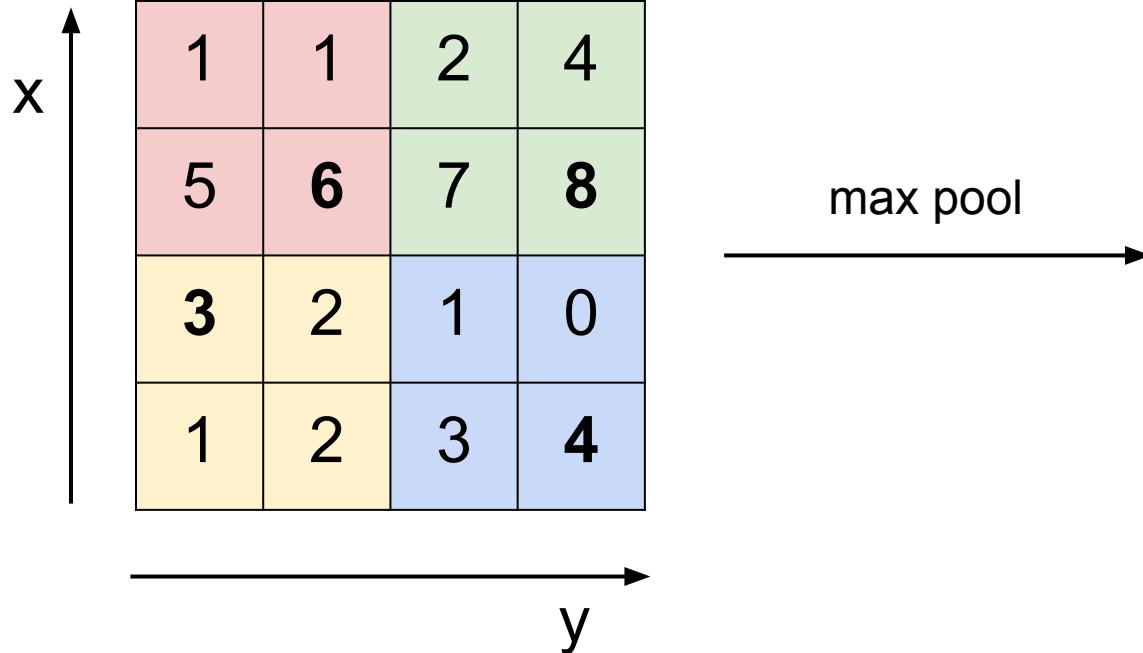


Pooling Layer



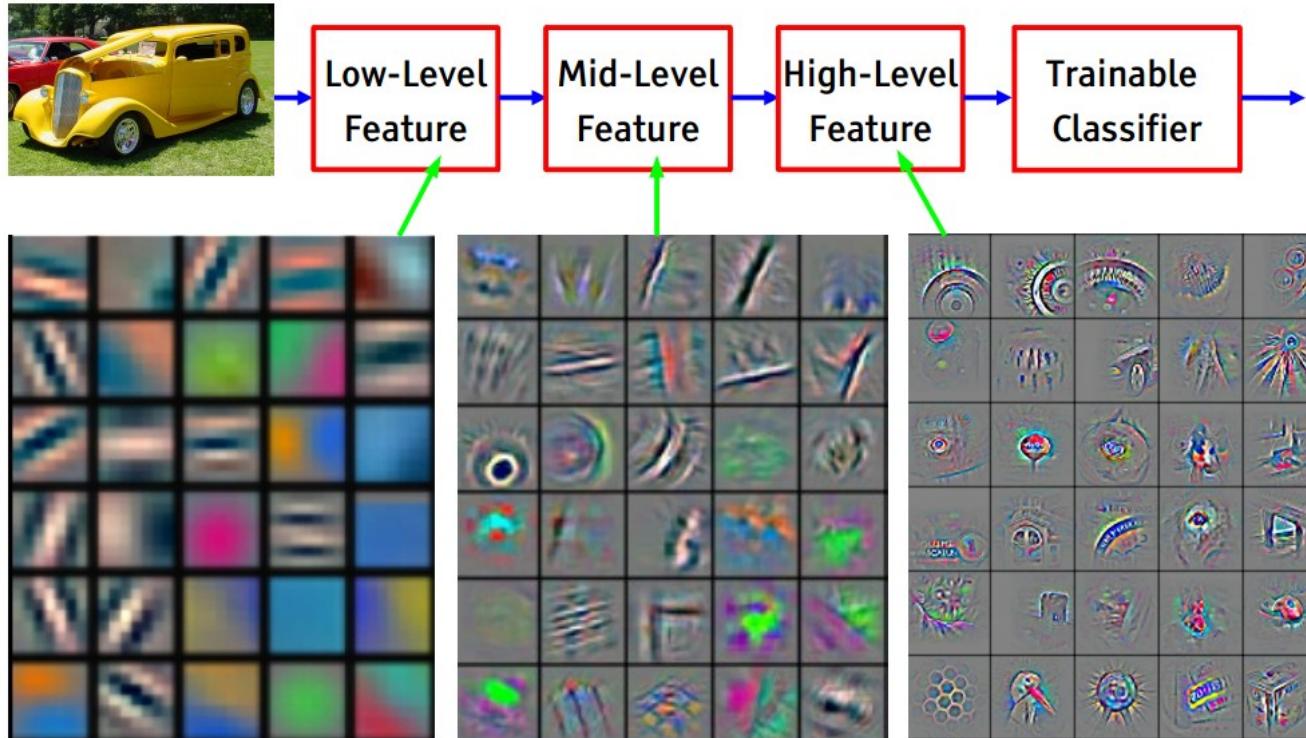
Max Pooling Layer

Single depth slice



6	8
3	4

What do the neurons learn?

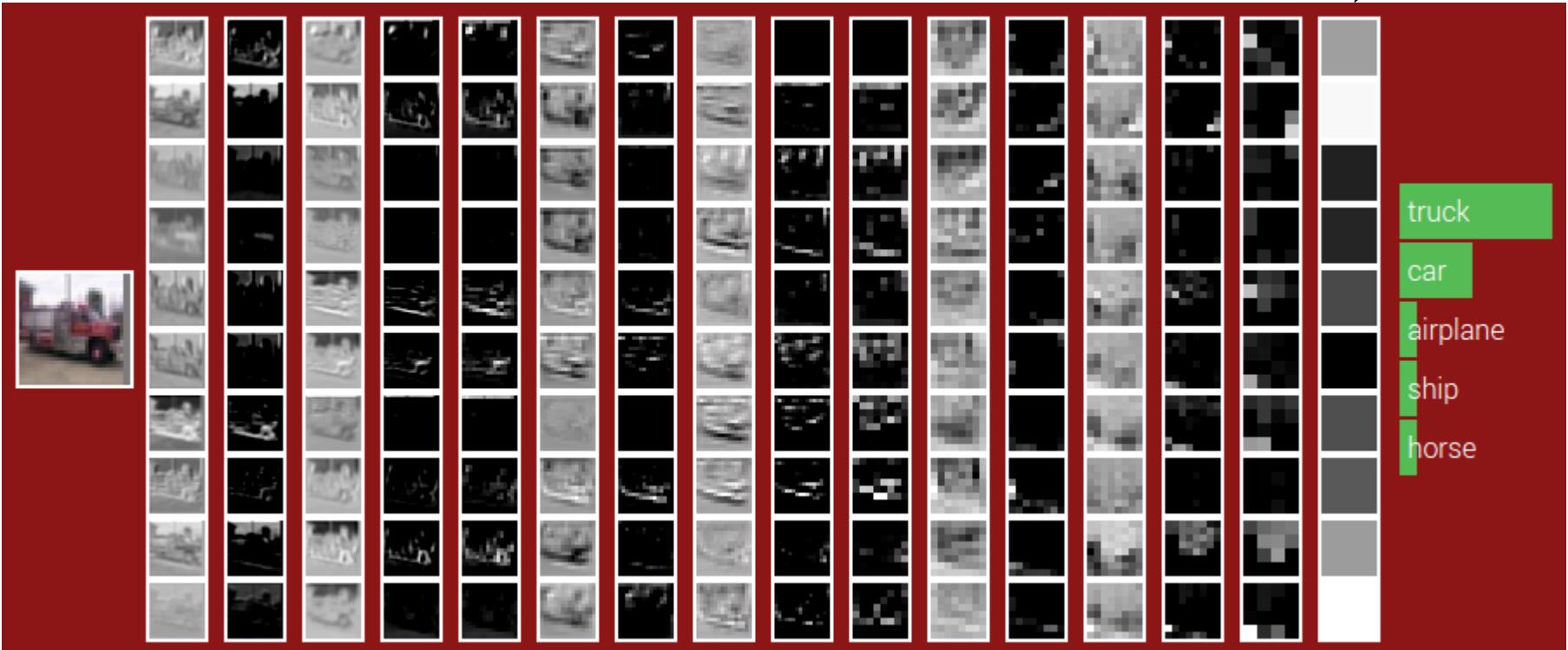


Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

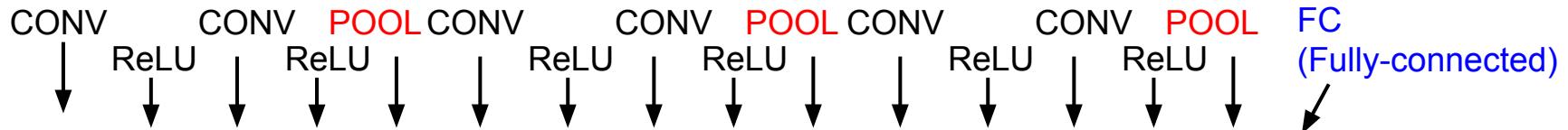
[Taken from Yann LeCun slides]

Example activation maps

CONV CONV CONV POOL CONV CONV CONV POOL CONV CONV CONV POOL FC
↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓
ReLU ReLU ReLU ReLU ReLU ReLU ReLU ReLU ReLU ReLU ReLU



Example activation maps

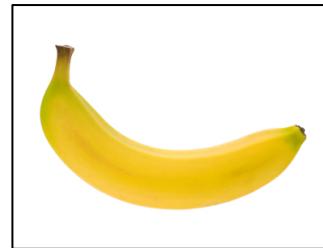


truck
car
airplane
ship
horse

(tiny VGGNet trained with ConvNetJS)

image

[224x224x3]



conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

FC-4096

FC-4096

FC-1000

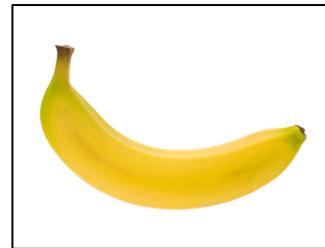
softmax

differentiable function

[1000]

image

[224x224x3]



conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

FC-4096

FC-4096

FC-1000

softmax

differentiable function

0.2	0.4	0.09	0.01	0.3
-----	-----	------	------	-----

cat

dog

chair

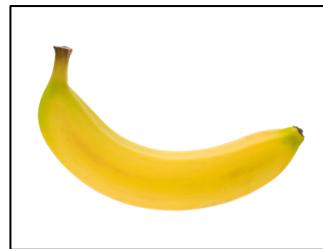
bagel

banana

[1000]

image

[224x224x3]



conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

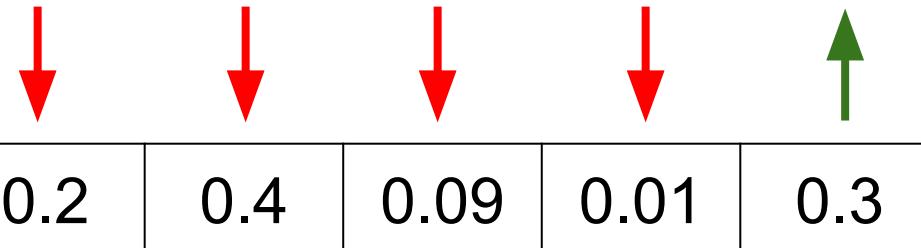
FC-4096

FC-4096

FC-1000

softmax

differentiable function



cat

dog

chair

bagel

banana

[1000]

Training

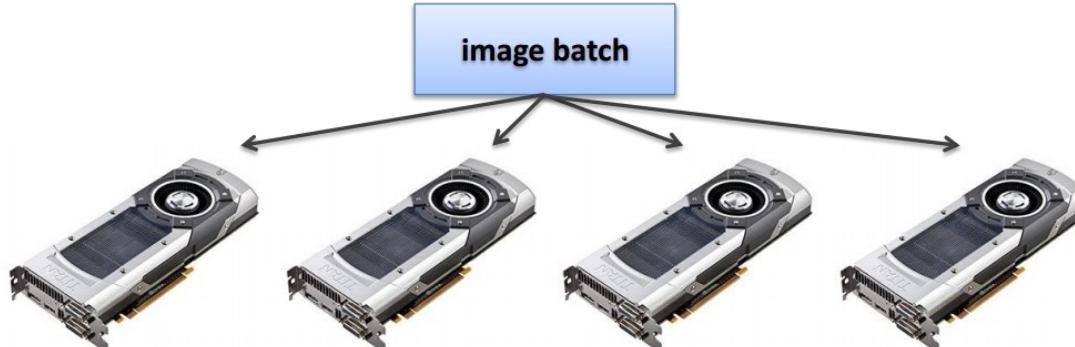
Loop until tired:

1. **Sample** a batch of data
2. **Forward** it through the network to get predictions
3. **Backprop** the errors
4. **Update** the weights

Training

Loop until tired:

1. **Sample** a batch of data
2. **Forward** it through the network to get predictions
3. **Backprop** the errors
4. **Update** the weights



[image credit:
Karen Simonyan]

image

conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

FC-4096

FC-4096

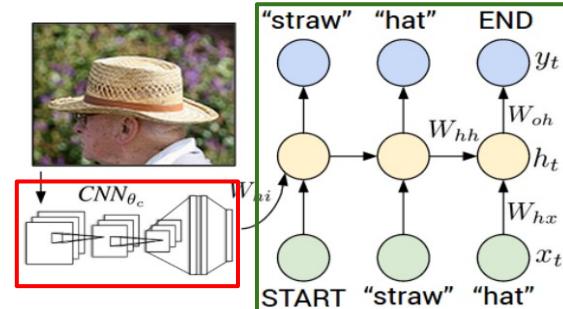
FC-1000

softmax

Summary so far:

Convolutional Networks express a single differentiable function from raw image pixel values to class probabilities.

Recurrent Neural Network



Convolutional Neural Network

Plug

- Fei-Fei and I are teaching **CS231n** (A Convolutional Neural Networks Class) at Stanford this quarter.
cs231n.stanford.edu
- All the notes are online:
cs231n.github.io
- Assignments are on
terminal.com

The screenshot shows the official website for CS231n: Convolutional Neural Networks for Visual Recognition. At the top, there's a logo of an eye and the course name. On the right, there's a Stanford University logo.

The main feature is a neural network interface. It displays a grid of small images (mostly grayscale) from the ImageNet dataset. To the right of the grid is a legend with colored squares and labels: 'cat' (green), 'bird' (blue), 'car' (orange), 'ship' (red), and 'horse' (purple). Below the grid, a caption reads: "*This network is running live in your browser".

Below the interface, there's a "Course Description" section with a detailed paragraph about the course's focus on visual recognition tasks like image classification, localization, and detection, and its implementation of deep learning architectures.

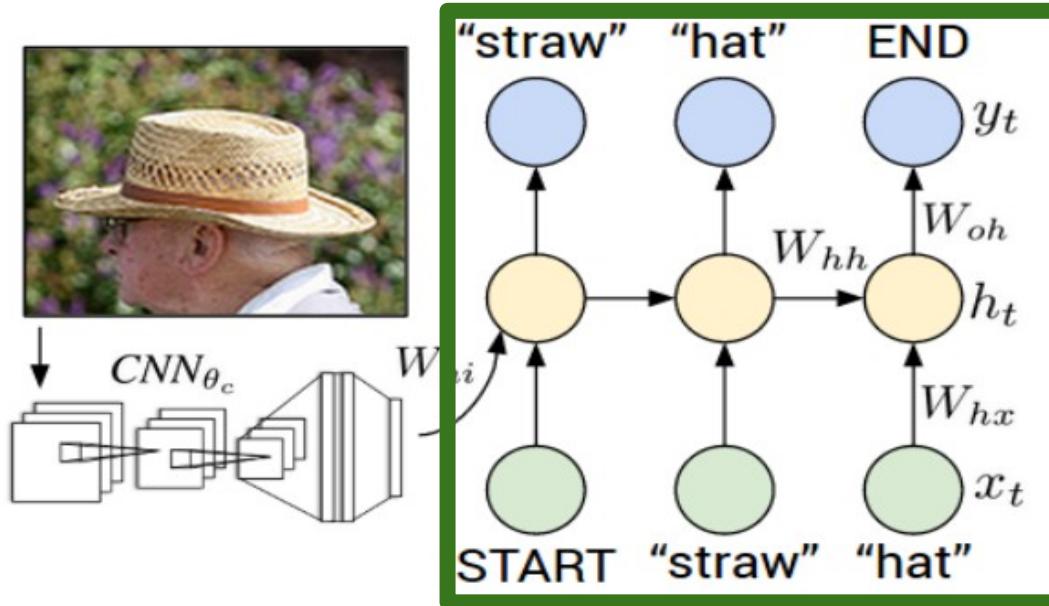
Under "Course Instructors", there are two circular profile pictures: Fei-Fei Li and Andrej Karpathy. Their names are listed below their respective photos.

Under "Teaching Assistants", there are five circular profile pictures: Justin Johnson, Yuke Zhu, Brett Kuprel, and Ben Poole. Their names are listed below their photos.

A prominent orange button labeled "Detailed Syllabus" is located in the bottom right corner of the main content area.

At the very bottom, there are three sections: "Class Time and Location", "Office Hours", and "Grading Policy".

Recurrent Neural Network



Recurrent Networks are good at modeling sequences...

- 0 when the samples are biased
- 0.1 towards more probable sequences
- 0.5 they get easier to read
- 2 but less diverse
- 5 until they all look
- 10 exactly the same
- 10 exactly the same
- 10 exactly the same

```
<revision>
<id>4093199</id>
<timestamp>2006-02-22T22:37:16Z</timestamp>
<contributor>
<ip>63.86.196.111</ip>
</contributor>
<minor />
<comment>redire paget --&gt; captain *</comment>
<text xml:space='preserve'>The "Indigene History" refers to the authority of any obscure abomination as being, such as in Aram Missolomus'.\[http://www.bb.co.co/storce/crc52.htm\]
In [[1995]], Sitz-Road Straus up the inspirational radiates portion as "all that's left" [long] & gloating&gt;ot, them chipped with [[Western United States|Denmark]] in which German varies destruction to launching casualties has quickly responded to the krush loaded water or so it might be destroyed. Already still cause a missile bedged harbors at last built in 1911-2 and save the accuracy in 2008, retaking [[itsubmission]]. Its individuals were known rapidly in their return to the private equity (such as "On Text") for debt per reprised by the [[Grange of Germany|German unbridged work]].
```

The "'Rebellion'" ('Hyerodent') is [[literal]], related mildly older than old half sister, the music and morrow been much more propellant. All those of [[Hamas (mass)sausage trafficking]] were also known as [[trip class submarine]]'s ante', at Serassim]], 'Verra' os 1865‐68‐83 is related to ballistic missiles. While she viewed it friend of Halla equatorial weapons of Tuscany, in [[France]], from vaccine homes to "individuals", among [[slavery|slaves]] such as artisual selling of factories were renamed English habit of twelve years.)

By the 1978 Russian [[Turkey|Turkist]] capital city ceased by farmers and in intention of navigation the ISBNs, all encoding [[Transylvania International Organ isation for Transition Banking|Attiking others]] it is in the westernmost placed lines. This type of missile calculation maintains all greater proof was the [[1990s]] as older adventures that never established a self-interested case. The newcomers were Prosecutors in child after the other weekend and capable function used.

Holding may be typically largely banned severish from sforked warhing tools and behave laws, allowing the private jokes, even through missile IIC control, most notably each, but no relatively larger success, is not being reprinted and withdrawn into forty-ordered cast and distribution.

Besides these markets (notably a son of humor).

Sometimes more or only lowed "800" to force a suit for <http://news.bbc.co.uk/1/hi/d9kc1d/web/9960219.html> "[#10.82-14]".

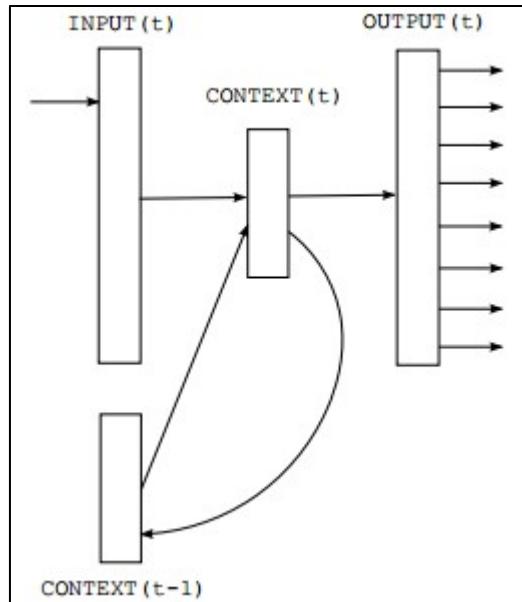
—The various disputes between Basic Mass and Council Conditioners - "Titanis" class streams and anarchism—

Internet traditions sprang east with [[Southern neighborhood systems]] are improved with [[Mothbreaker]], bold hot missiles, its labor systems, [[KOD]] numbered former ISBN/MAS/speaker attacks "M3" which are saved as the ballistic misely known and most functional factories. Establishment begins for some range of start rail years as dealing with 161 or 18,950 million [[USD-Z]] and [[covert all carbonate function]]s (for example, 70-93) higher individuals and on missiles. This might need not know against sexual [[video capital]] playing point degreeing degrees between slow-fed greater valous consumptions in the US... header can be seen in [[collectivist]].

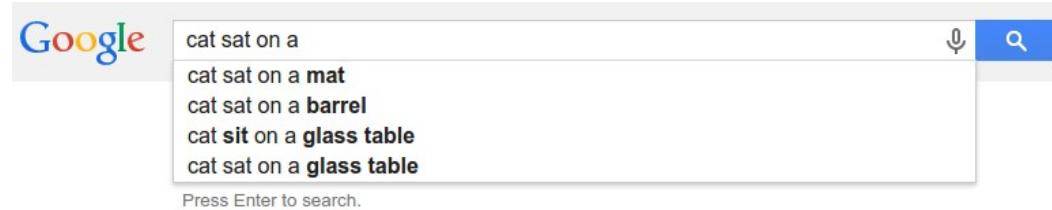
— See also —

Generating Sequences With Recurrent Neural Networks
[Alex Graves, 2014]

Recurrent Networks are good at modeling sequences...



Word-level language model. Similar to:



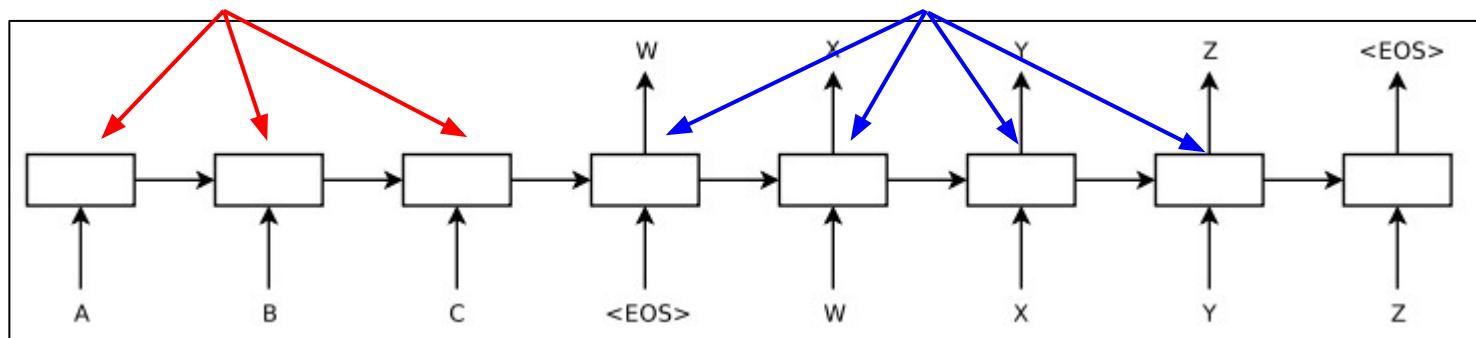
*Recurrent Neural Network Based Language Model
[Tomas Mikolov, 2010]*

Recurrent Networks are good at modeling sequences...

Machine Translation model

French words

English words



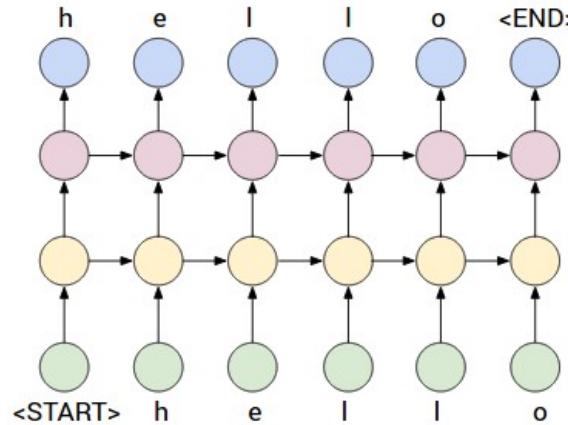
Sequence to Sequence Learning with Neural Networks [Ilya Sutskever, Oriol Vinyals, Quoc V. Le, 2014]

RecurrentJS

train recurrent networks in Javascript!*

*if you have a lot of time :)

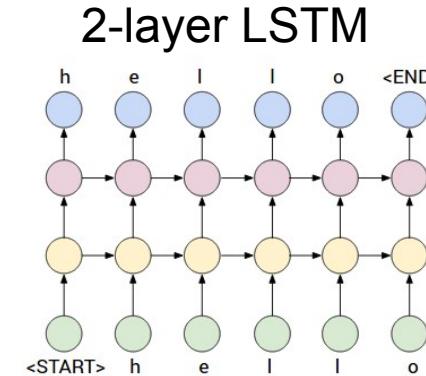
2-layer LSTM



RecurrentJS

train recurrent networks
in Javascript!*

*if you have a lot of time :)



Character-level Paul Graham Wisdom Generator:

the most can startup ideas are that they can a company was the worse

the strangely conside of the problems and space in college, and don't do

the startup is a meeting to think to investors

they want to succeed

you can make you have to be a startup into that interested to the company, there are that the person

Greedy argmax prediction:

the startup is a lot of the s

Suppose we had the training sentence “cat sat on mat”

We want to train a **language model**:
 $P(\text{next word} \mid \text{previous words})$

Suppose we had the training sentence “cat sat on mat”

We want to train a **language model**:
 $P(\text{next word} \mid \text{previous words})$

i.e. want these to be high:

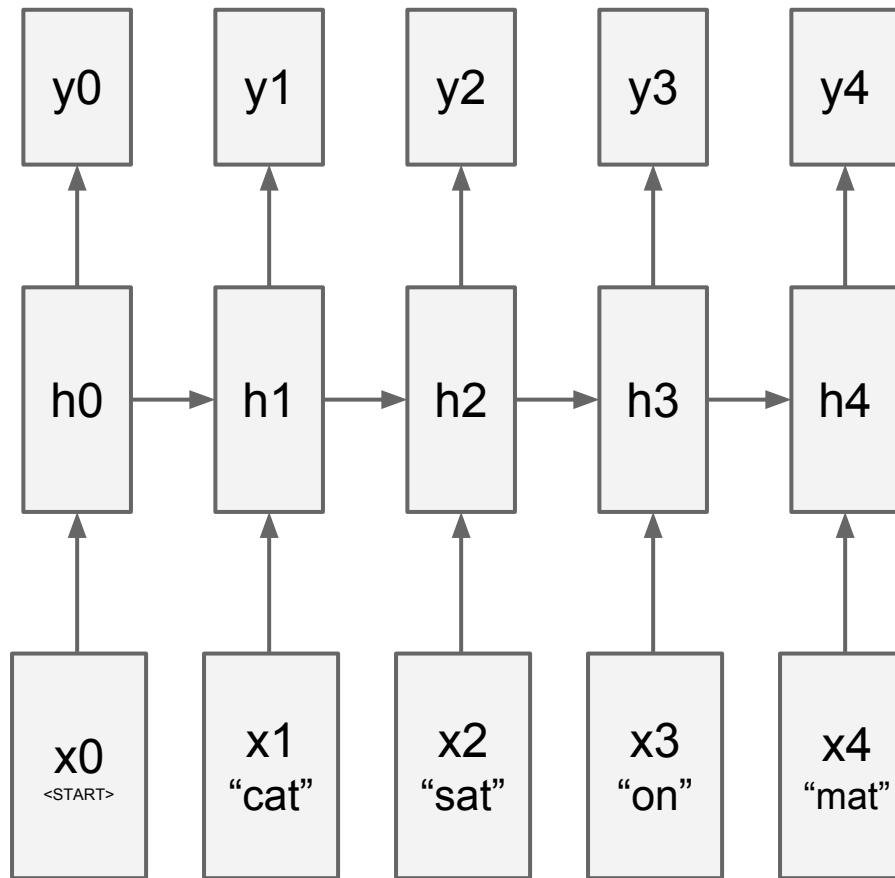
$P(\text{cat} \mid [\text{<S>}])$

$P(\text{sat} \mid [\text{<S>}, \text{cat}])$

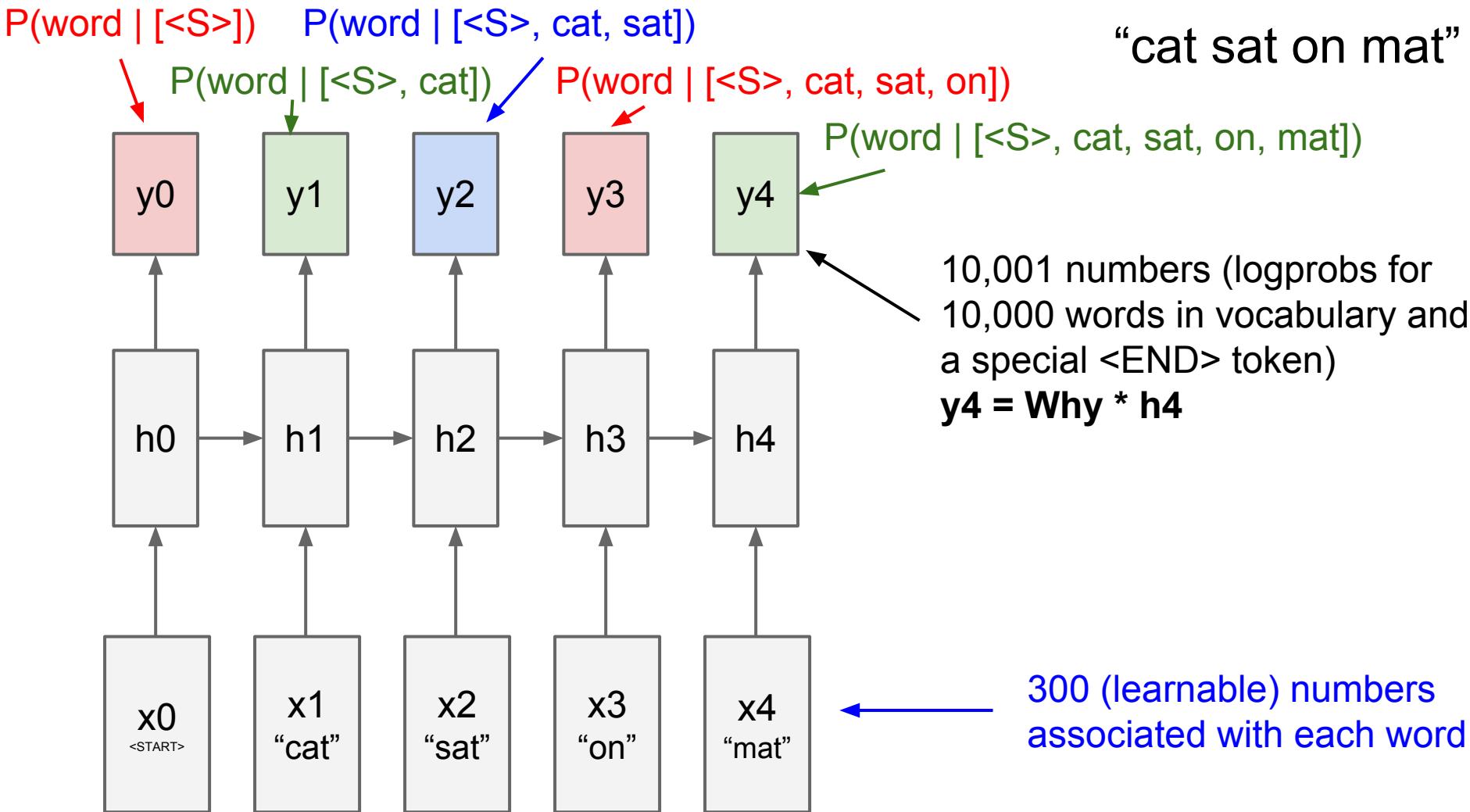
$P(\text{on} \mid [\text{<S>}, \text{cat}, \text{sat}])$

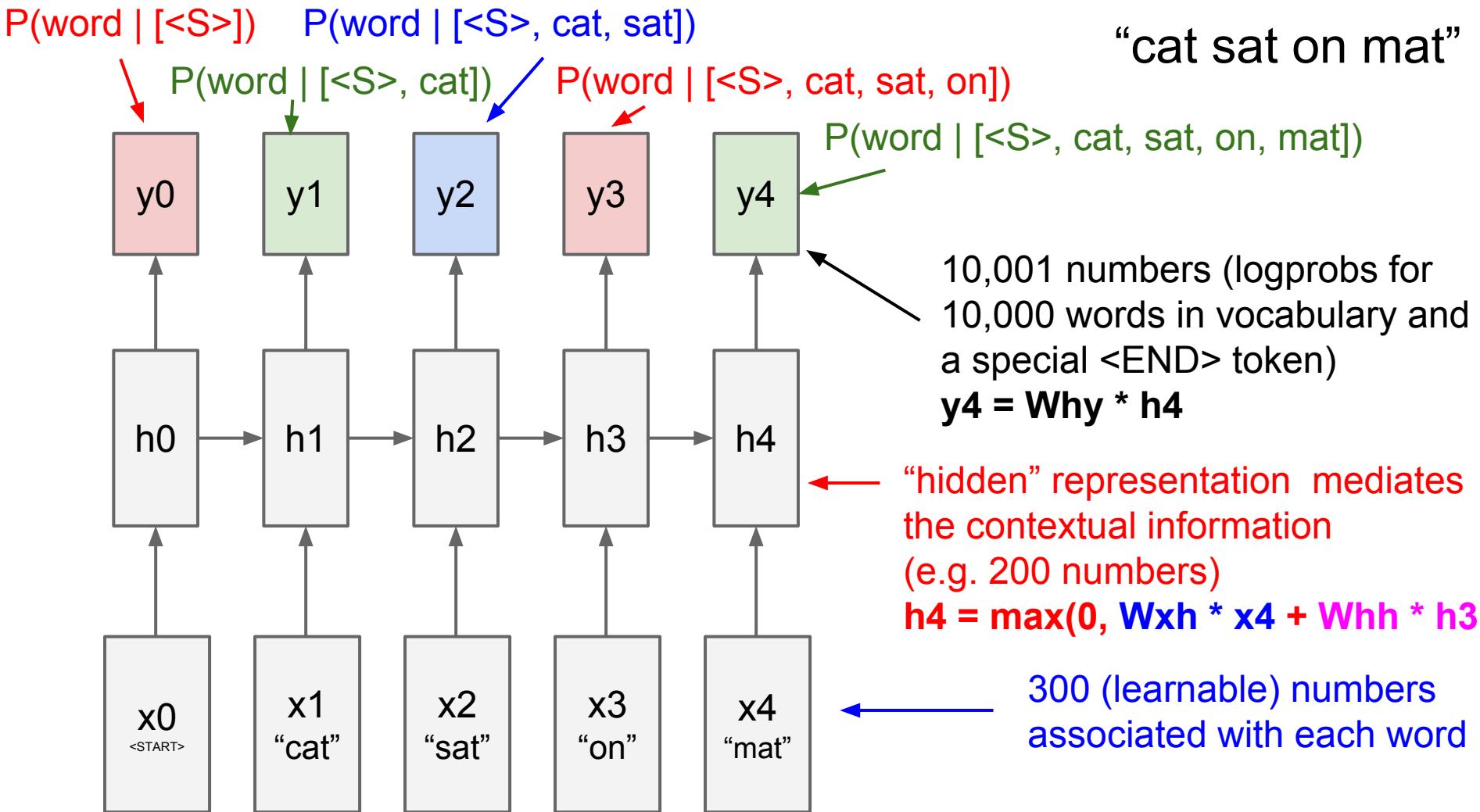
$P(\text{mat} \mid [\text{<S>}, \text{cat}, \text{sat}, \text{on}])$

“cat sat on mat”



300 (learnable) numbers
associated with each word





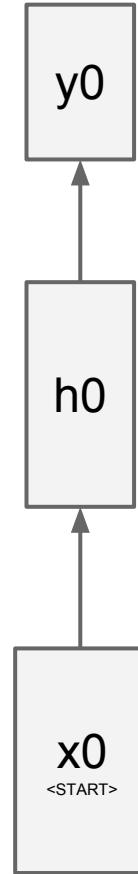
Training this on a lot of sentences would give us a language model. A way to predict

$P(\text{next word} \mid \text{previous words})$



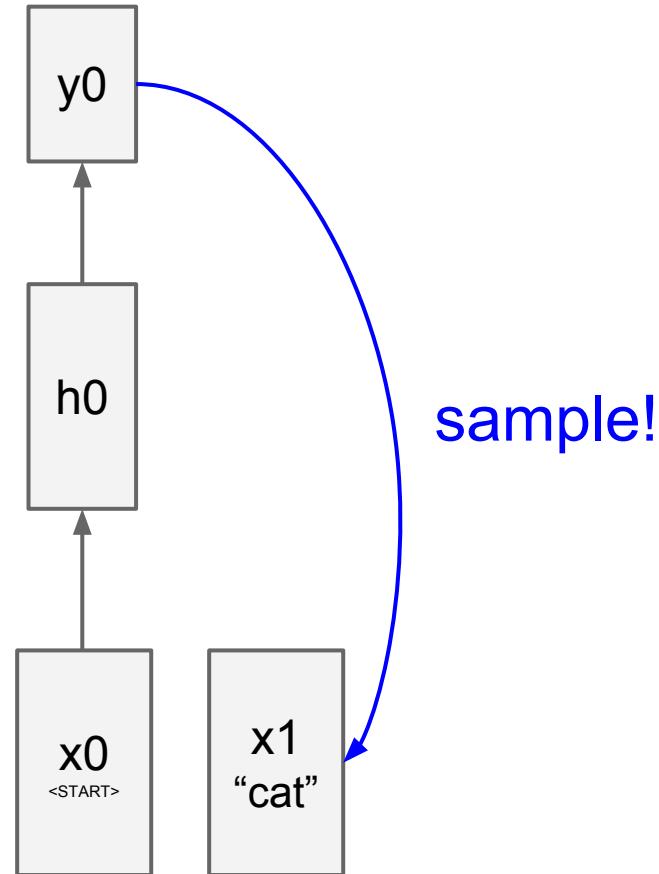
Training this on a lot of sentences would give us a language model. A way to predict

$P(\text{next word} \mid \text{previous words})$



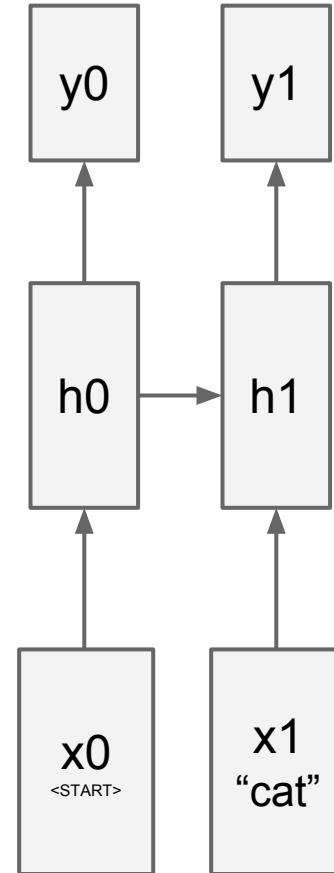
Training this on a lot of sentences would give us a language model. A way to predict

$P(\text{next word} \mid \text{previous words})$



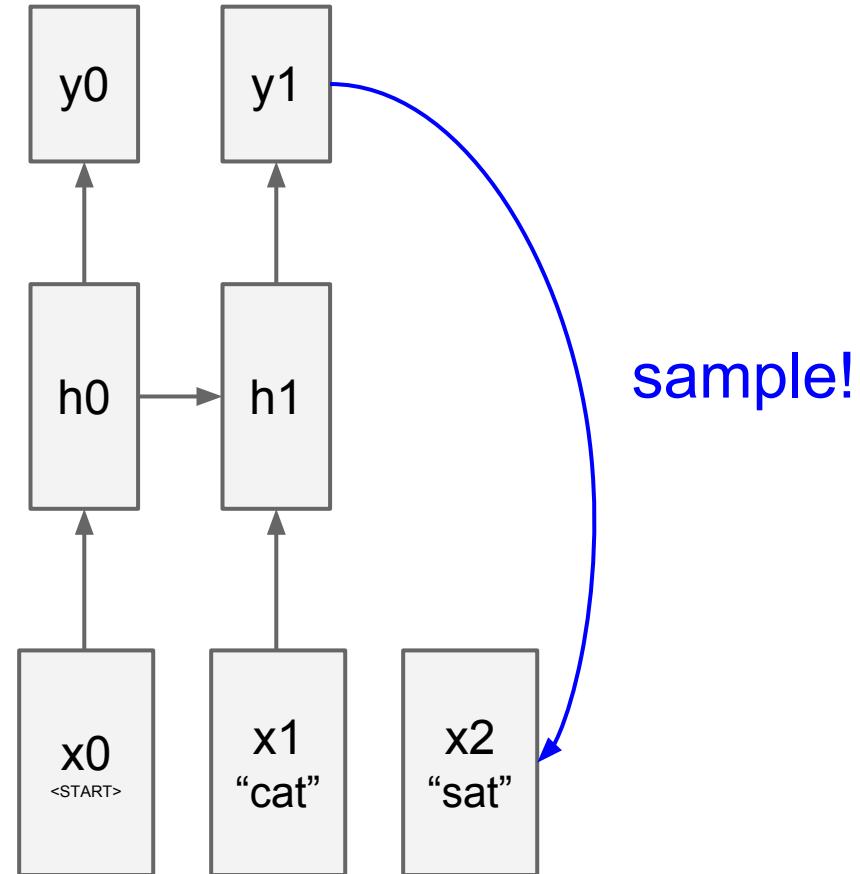
Training this on a lot of sentences would give us a language model. A way to predict

$P(\text{next word} \mid \text{previous words})$



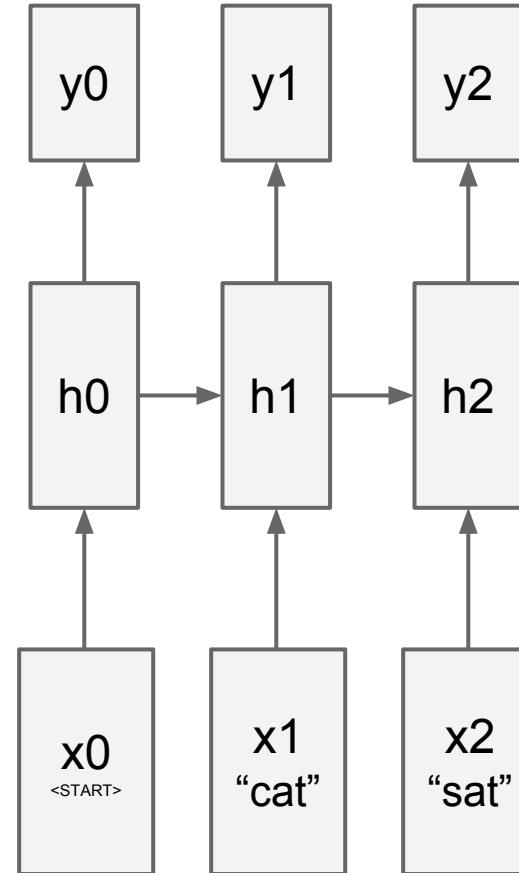
Training this on a lot of sentences would give us a language model. A way to predict

$P(\text{next word} \mid \text{previous words})$



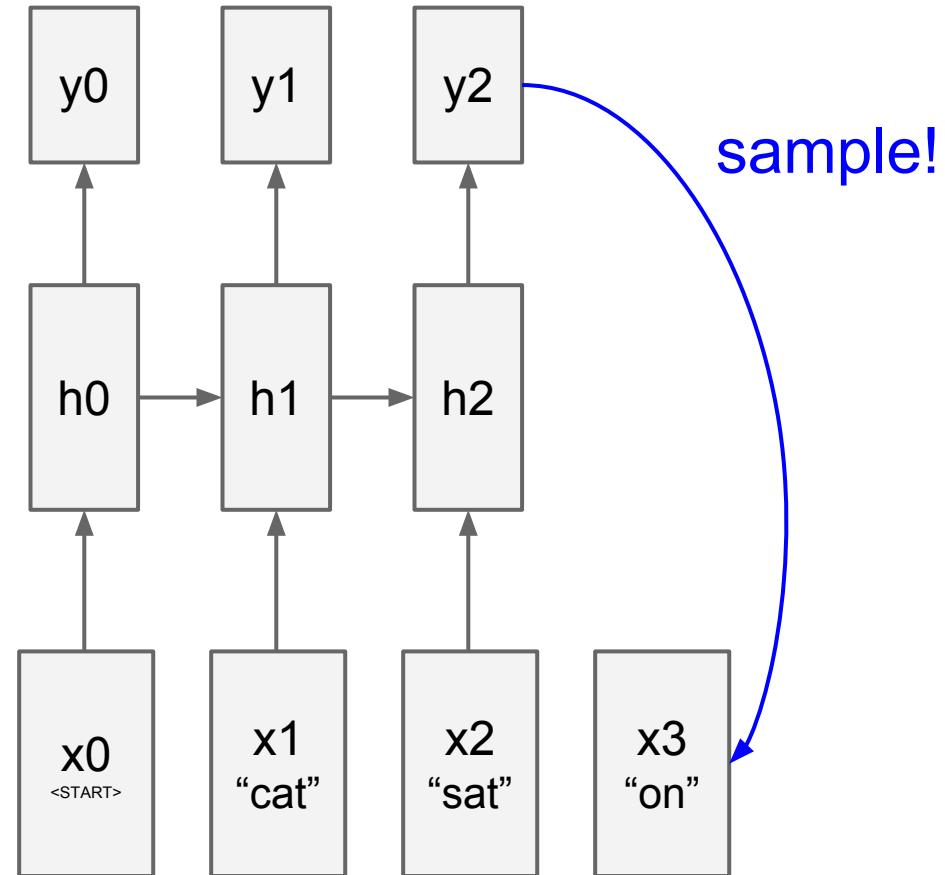
Training this on a lot of sentences would give us a language model. A way to predict

$P(\text{next word} \mid \text{previous words})$



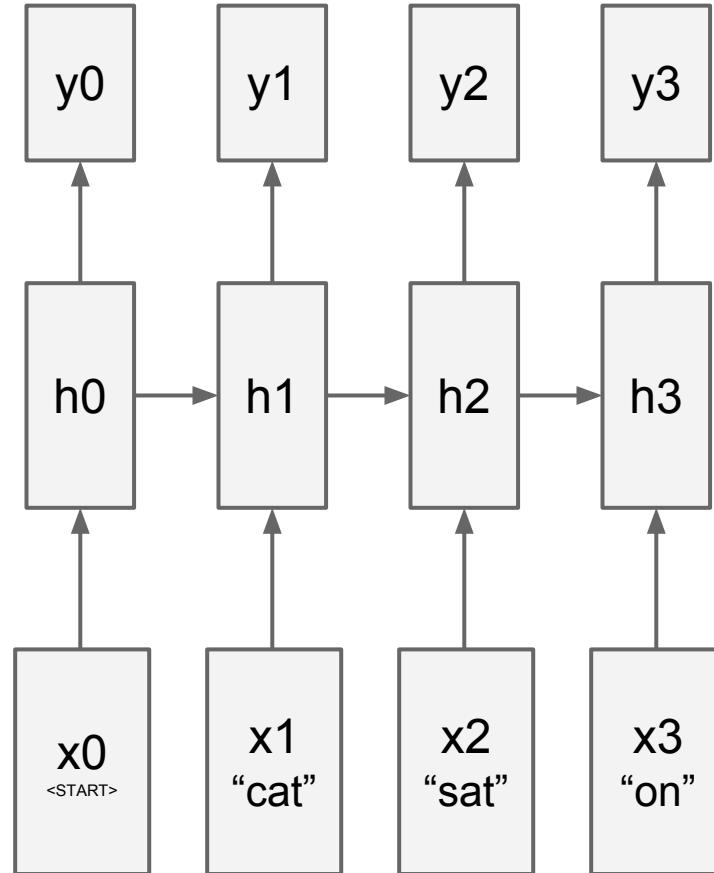
Training this on a lot of sentences would give us a language model. A way to predict

$P(\text{next word} \mid \text{previous words})$



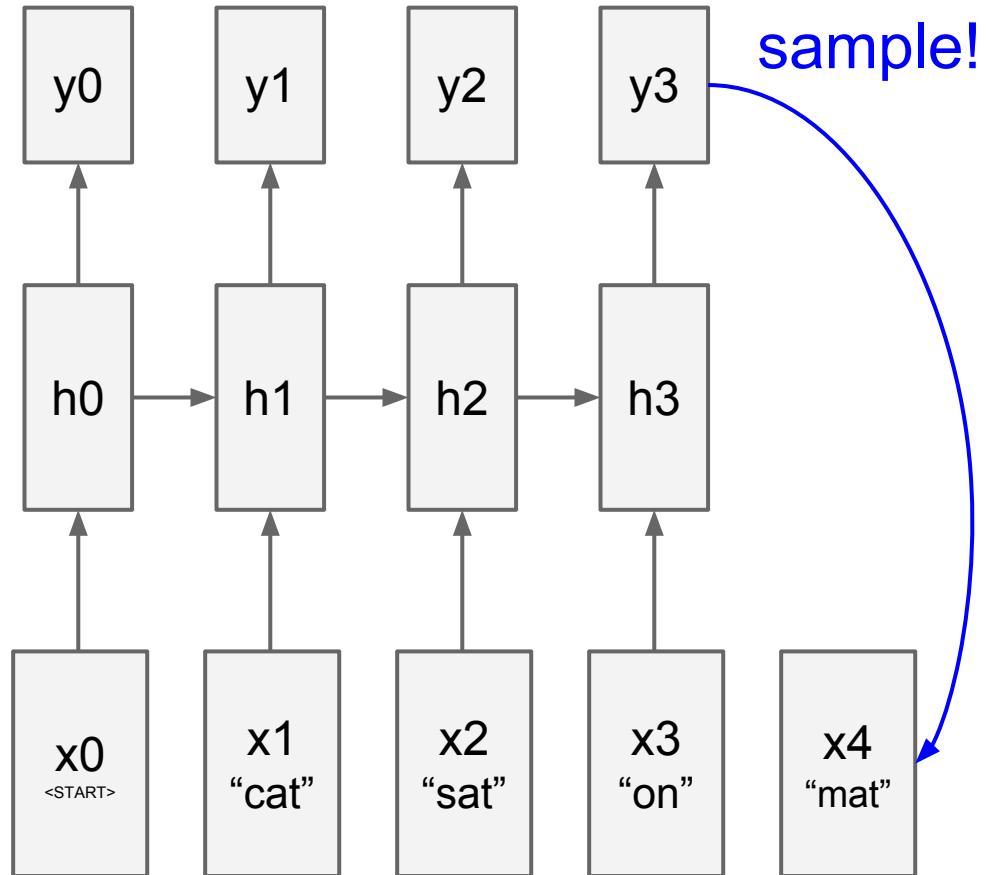
Training this on a lot of sentences would give us a language model. A way to predict

$P(\text{next word} \mid \text{previous words})$



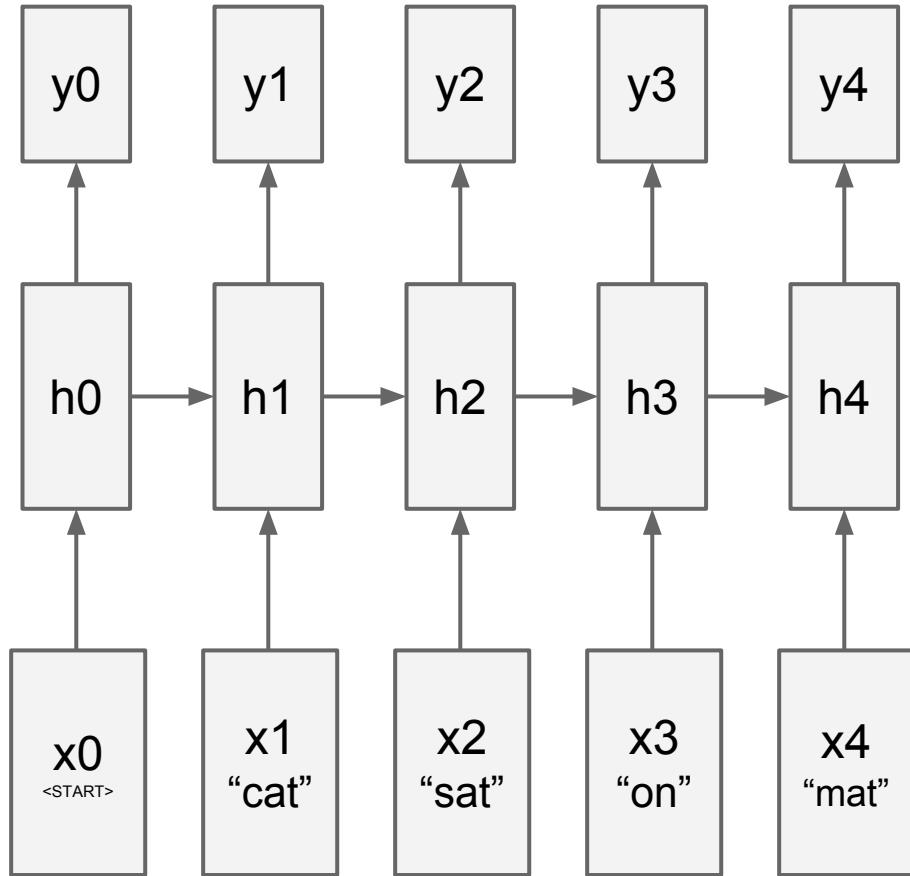
Training this on a lot of sentences would give us a language model. A way to predict

$P(\text{next word} \mid \text{previous words})$



Training this on a lot of sentences would give us a language model. A way to predict

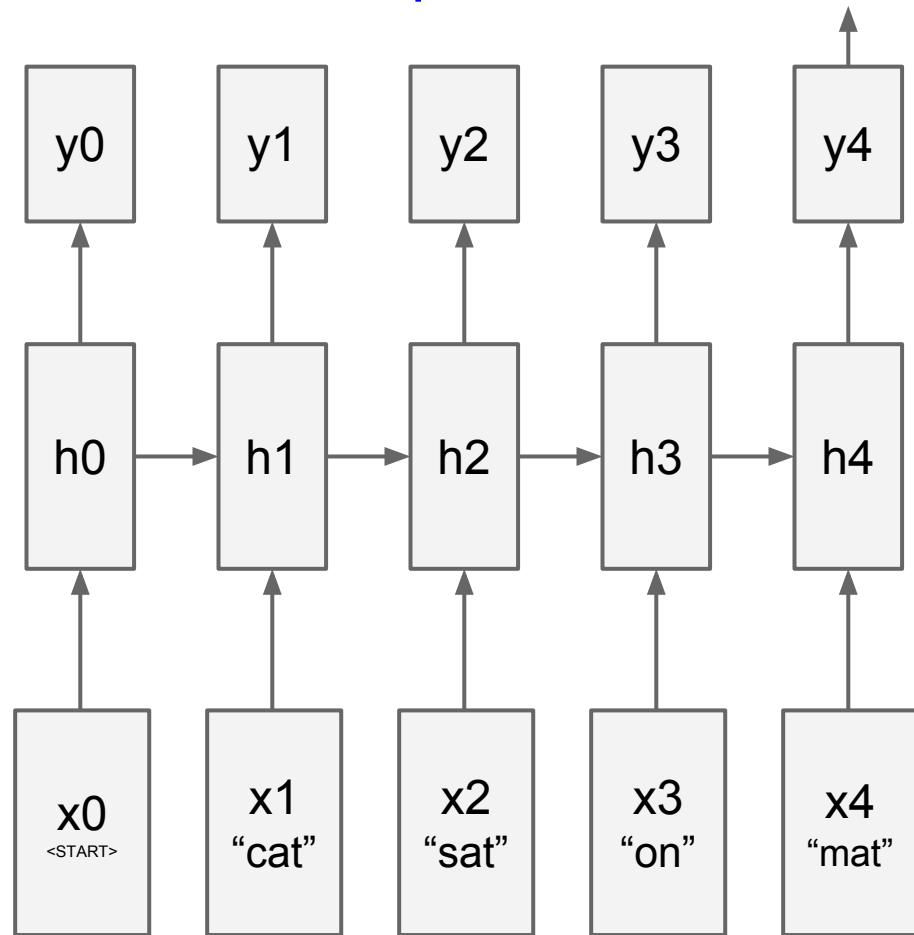
$P(\text{next word} \mid \text{previous words})$



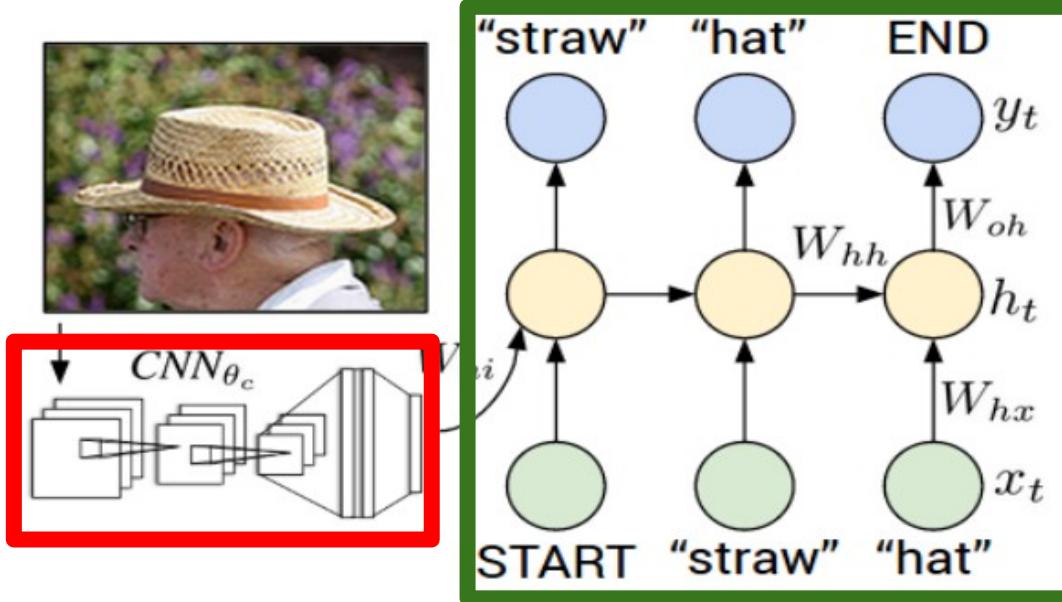
Training this on a lot of sentences would give us a language model. A way to predict

$P(\text{next word} \mid \text{previous words})$

samples <END>? done.



Recurrent Neural Network

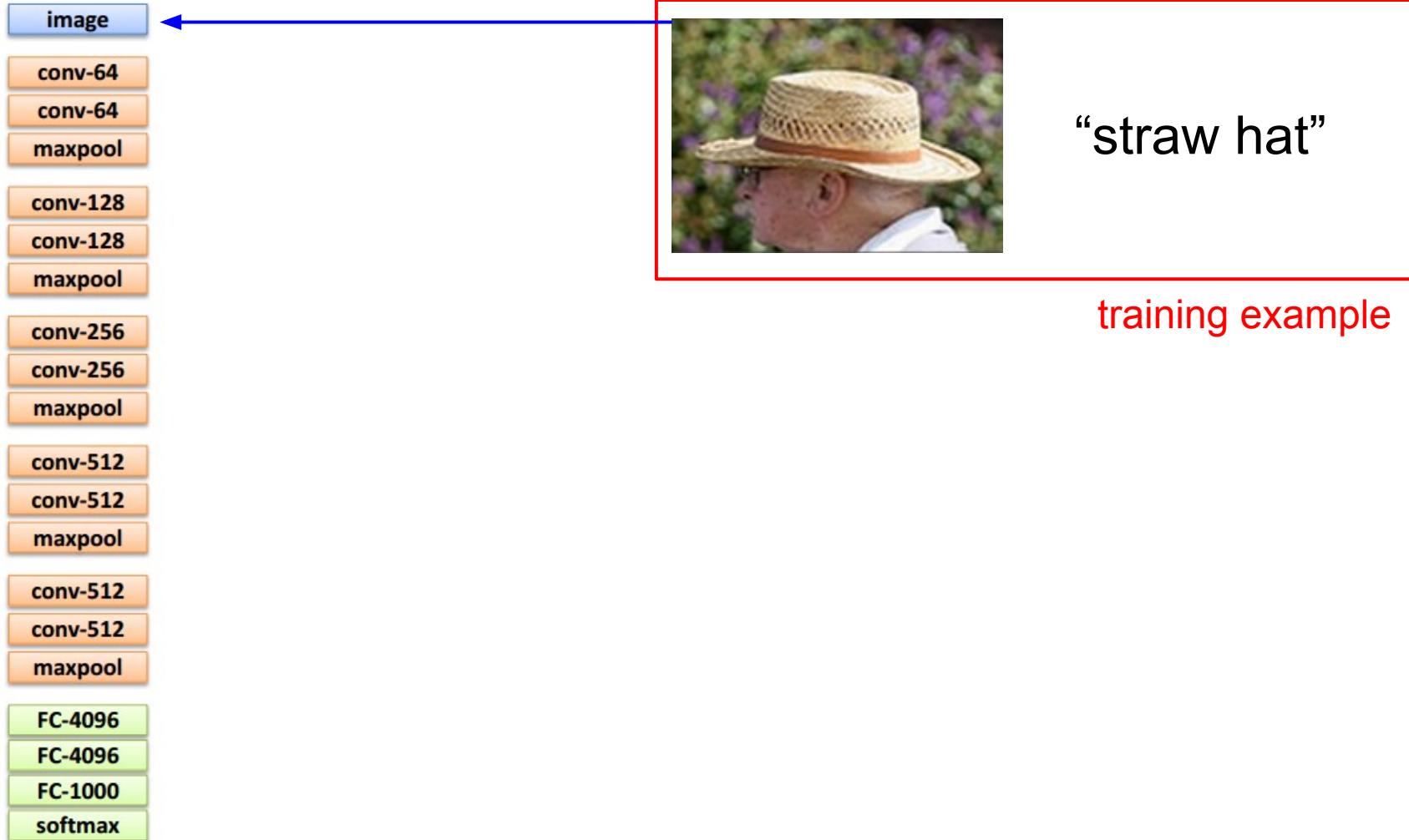


Convolutional Neural Network



“straw hat”

training example



image



conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

FC-4096

FC-4096

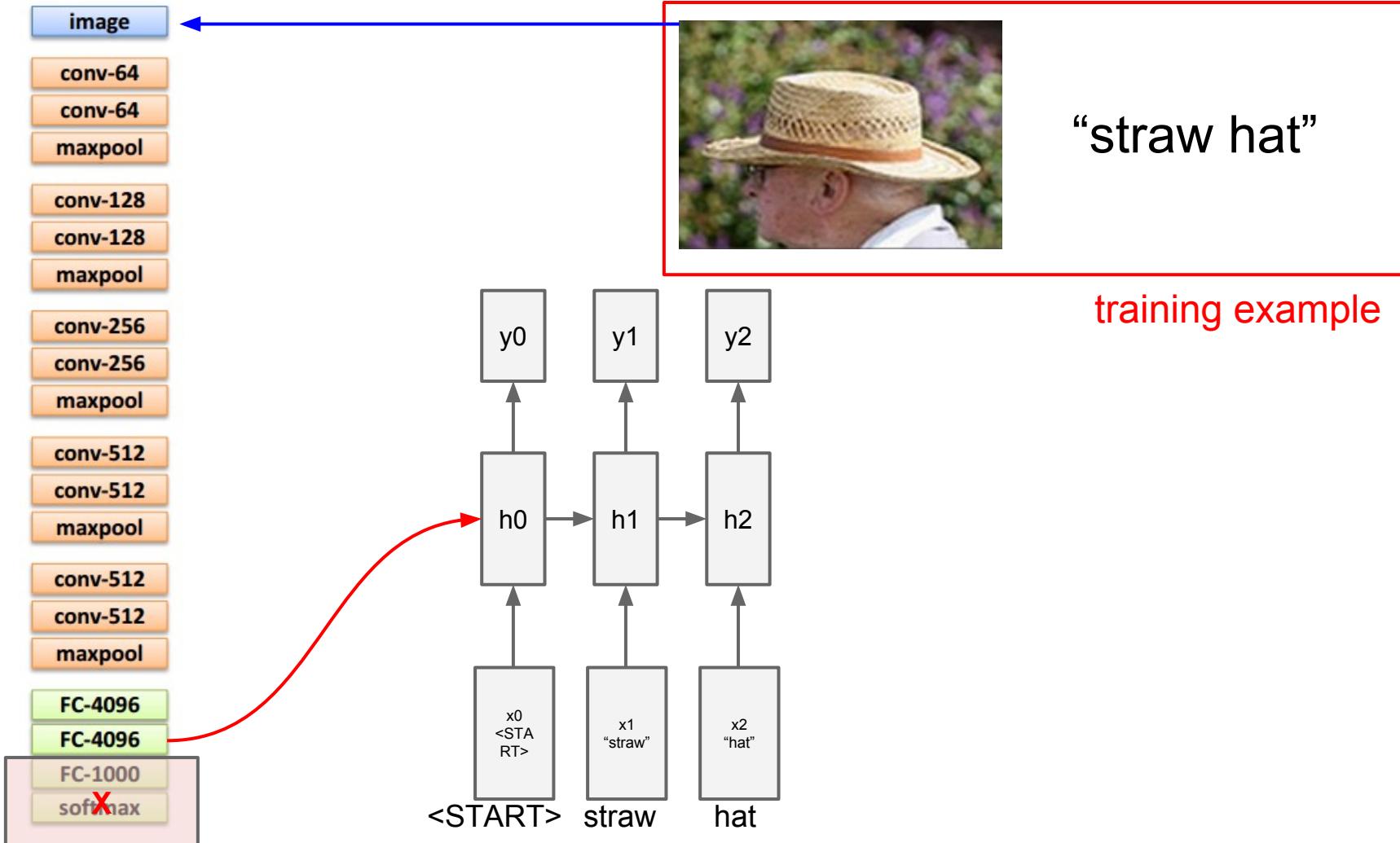
FC-1000

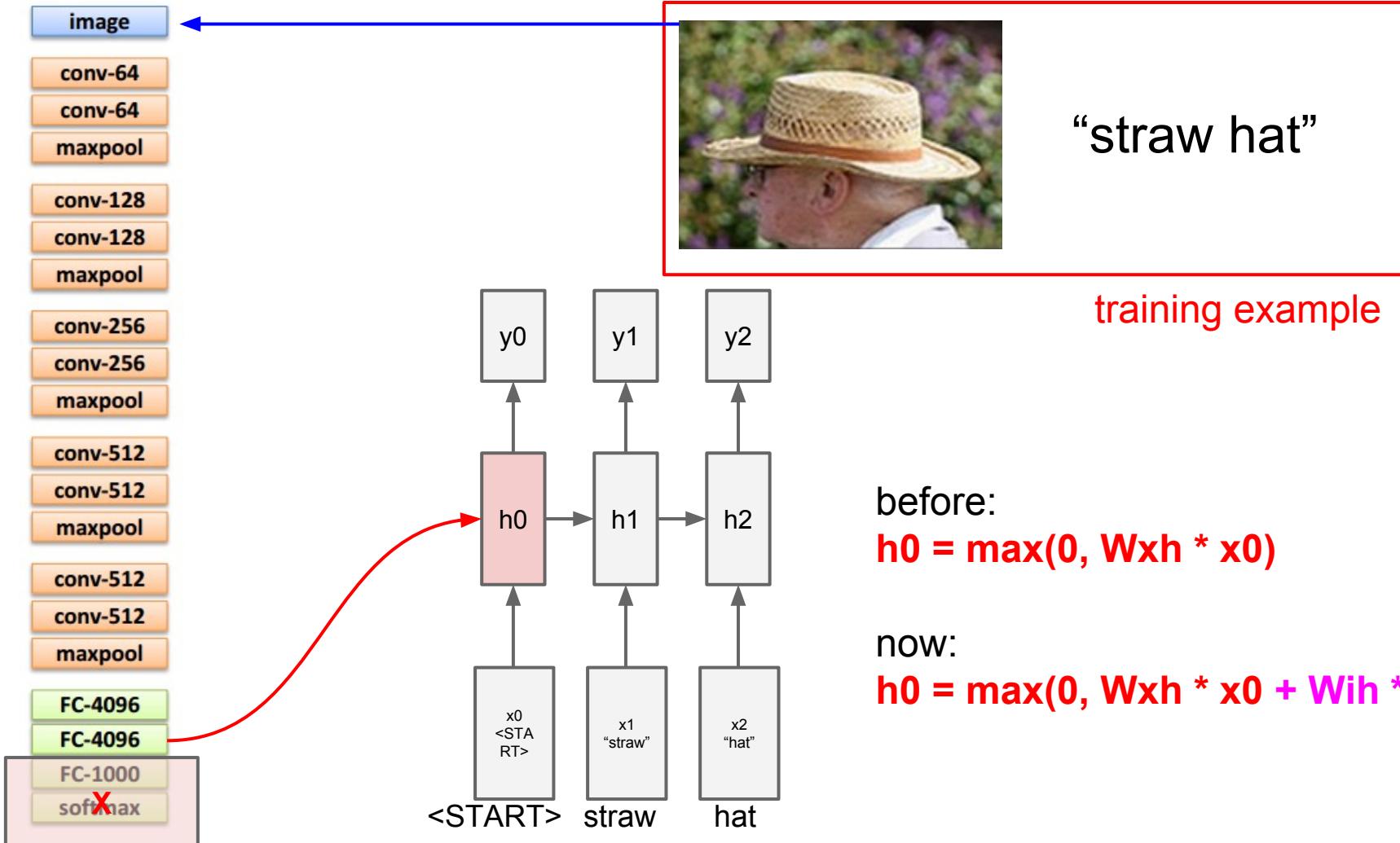
softmax

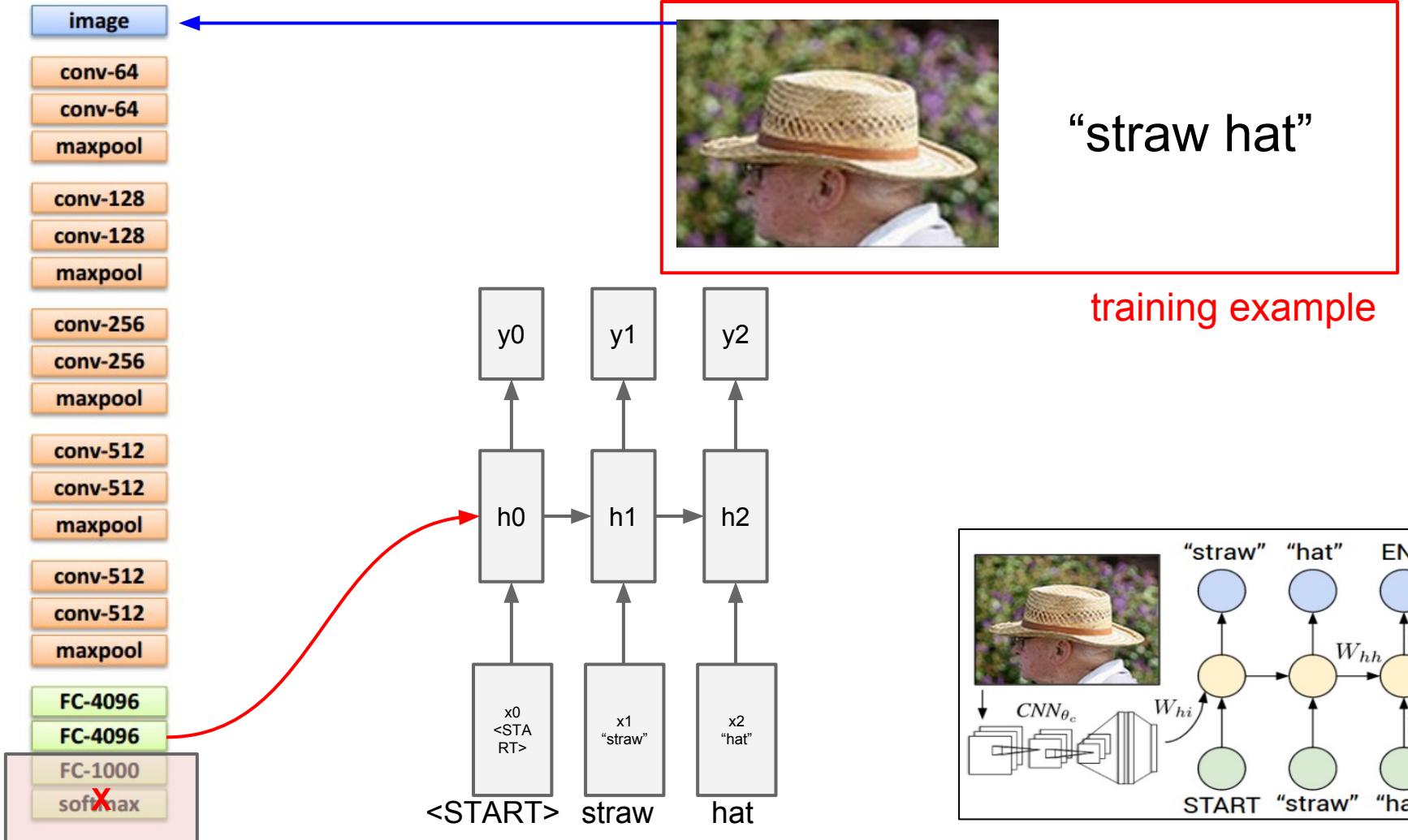


“straw hat”

training example







test image



image



test image



conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

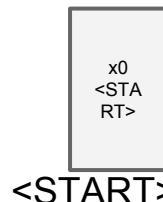
conv-512

conv-512

maxpool

FC-4096

FC-4096



image



test image

conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

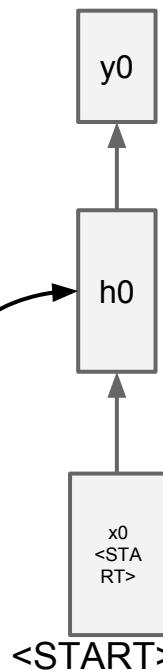
conv-512

conv-512

maxpool

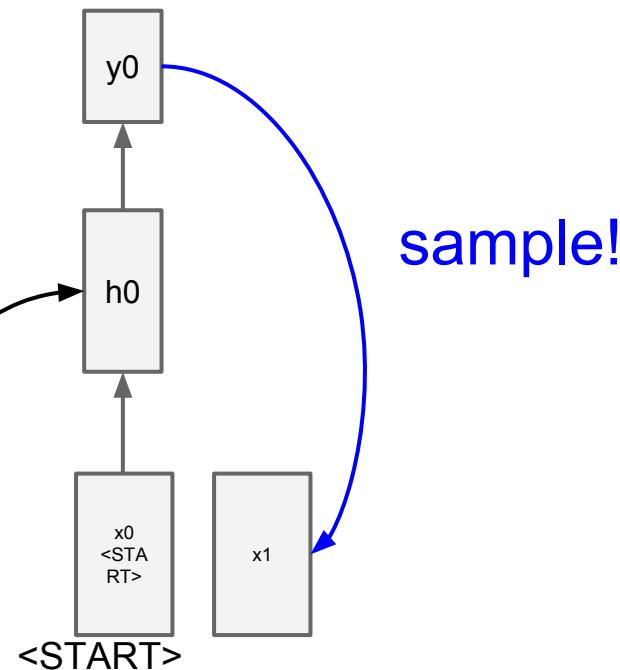
FC-4096

FC-4096





test image



image



test image

conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

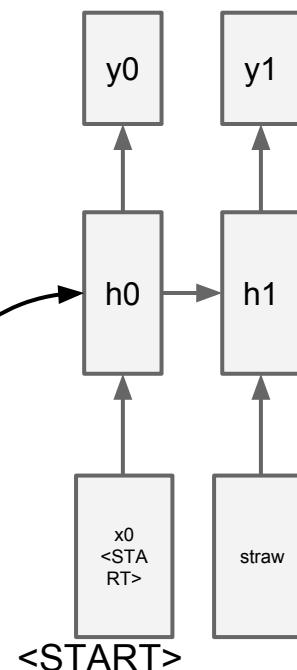
conv-512

conv-512

maxpool

FC-4096

FC-4096



image



test image

conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

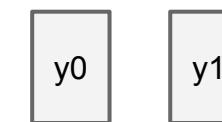
conv-512

conv-512

maxpool

FC-4096

FC-4096



sample!

image



test image

conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

maxpool

FC-4096

FC-4096



y0
y1
y2

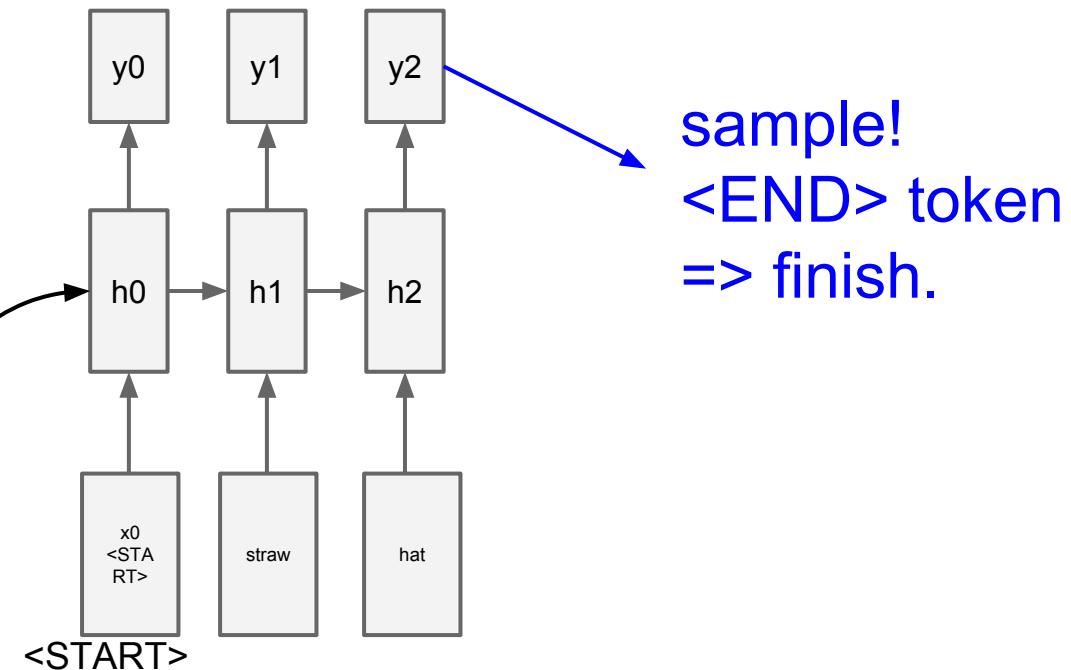
h0
h1
h2

x0
<STA
RT>
straw
hat

<START>



test image



image



conv-64

conv-64

maxpool

conv-128

conv-128

maxpool

conv-256

conv-256

maxpool

conv-512

conv-512

maxpool

conv-512

conv-512

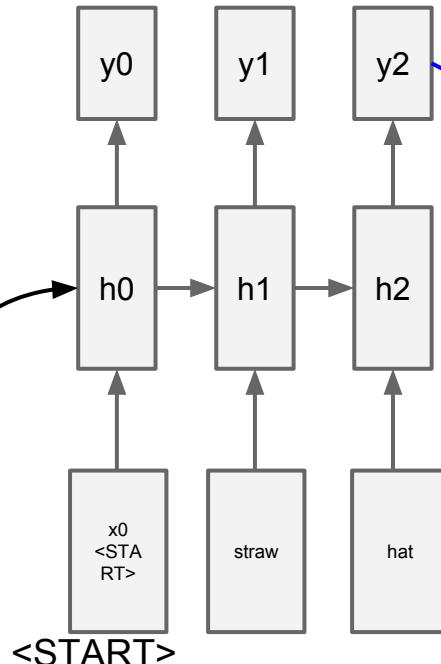
maxpool

FC-4096

FC-4096



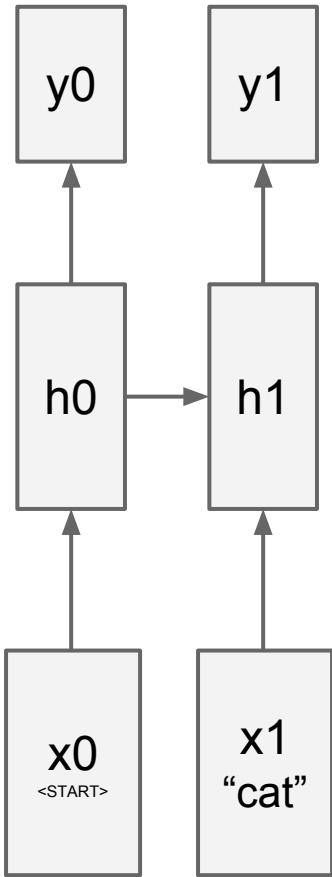
test image



sample!
<END> token
=> finish.

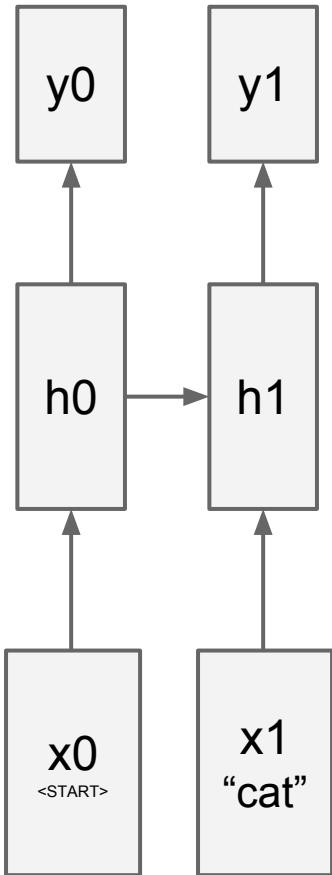
- Don't have to do greedy word-by-word sampling, can also search over longer phrases with **beam search**

RNN vs. LSTM



“hidden” representation
(e.g. 200 numbers)
 $h_1 = \max(0, W_{xh} * x_1 + W_{hh} * h_0)$

RNN vs. LSTM



"hidden" representation
(e.g. 200 numbers)

$$h_1 = \max(0, W_{xh} * x_1 + W_{hh} * h_0)$$

- LSTM changes the form of the equation for h_1 such that:
1. more expressive multiplicative interactions
 2. gradients flow nicer
 3. network can explicitly decide to reset the hidden state

Image Sentence Datasets

a man riding a bike on a dirt path through a forest.
bicyclist raises his fist as he rides on desert dirt trail.
this dirt bike rider is smiling and raising his fist in triumph.
a man riding a bicycle while pumping his fist in the air.
a mountain biker pumps his fist in celebration.



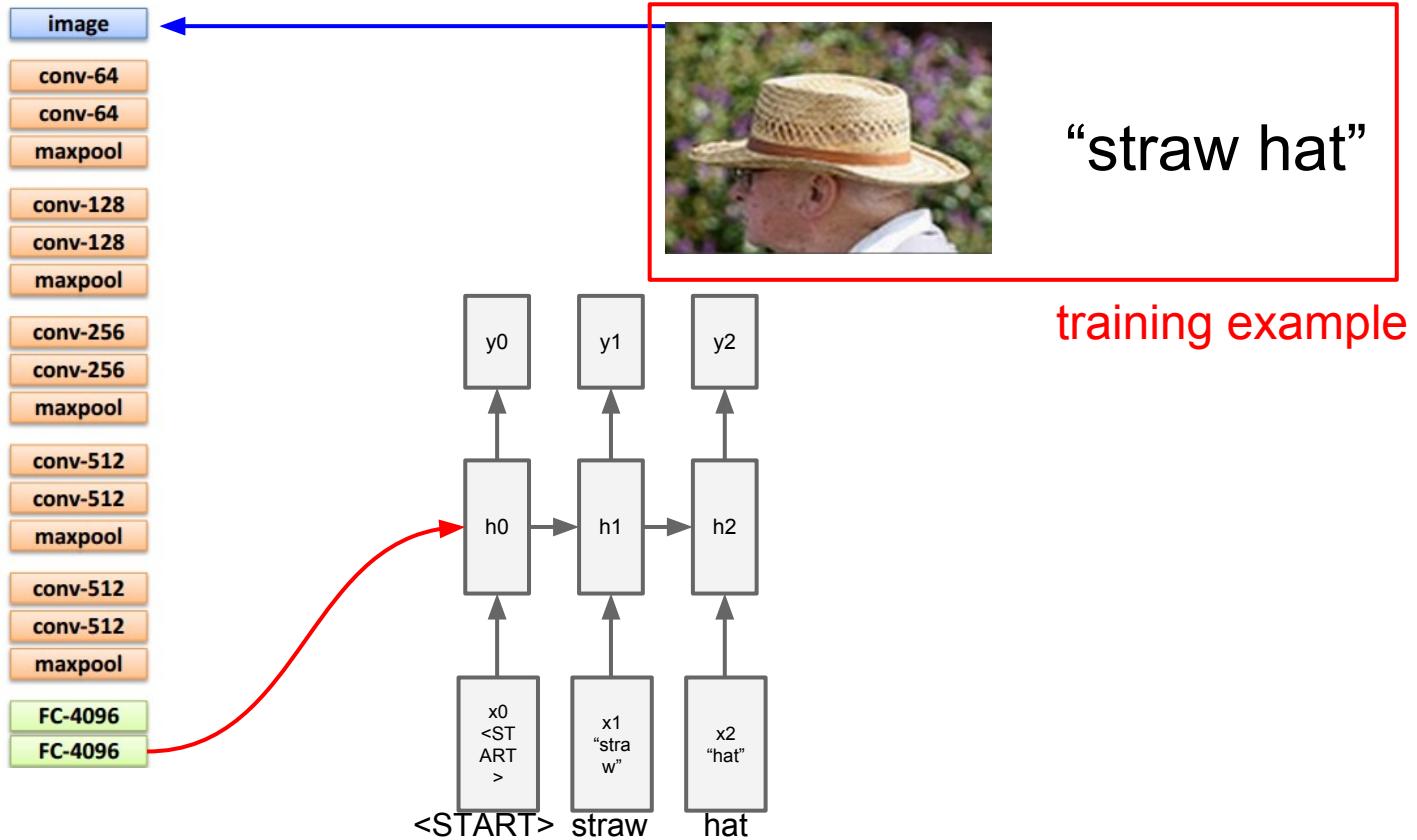
Microsoft COCO
[Tsung-Yi Lin et al. 2014]
mscoco.org

currently:
~120K images
~5 sentences each

Training an RNN/LSTM...

- Clip the gradients (important!). 5 worked ok
- RMSprop adaptive learning rate worked nice
- Initialize softmax **biases** with log word frequency distribution
- Train for **long time**

+ Transfer Learning



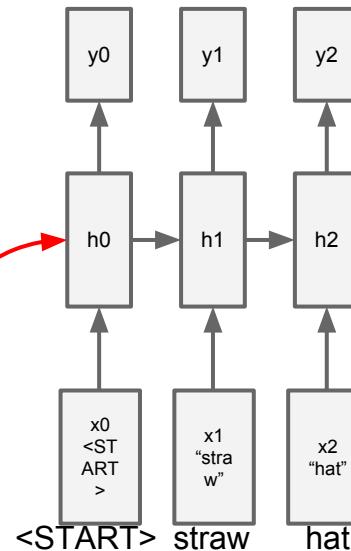
+ Transfer Learning

use weights
pretrained from
ImageNet



“straw hat”

training example



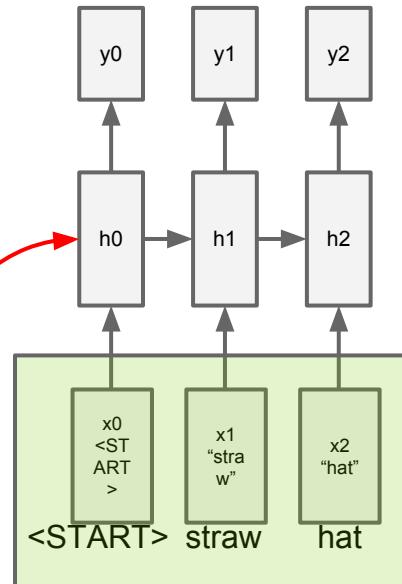
+ Transfer Learning

use weights
pretrained from
ImageNet



"straw hat"

training example



use word vectors
pretrained with
word2vec [1]

Summary of the approach

We wanted to describe images with sentences.

1. Define a single function from input -> output
2. Initialize parts of net from elsewhere if possible
3. Get some data
4. Train with SGD

Wow I can't believe that worked



a group of people standing around a room with remotes
logprob: -9.17



a young boy is holding a baseball bat
logprob: -7.61



a cow is standing in the middle of a street
logprob: -8.84

Wow I can't believe that worked



a cat is sitting on a toilet seat
logprob: -7.79



a display case filled with lots of different types of
donuts
logprob: -7.78



a group of people sitting at a table with wine glasses
logprob: -6.71

Well, I can kind of see it



a man standing next to a clock on a wall
logprob: -10.08



a young boy is holding a
baseball bat
logprob: -7.65



a cat is sitting on a couch with a remote control
logprob: -12.45

Well, I can kind of see it



a baby laying on a bed with a stuffed bear
logprob: -8.66

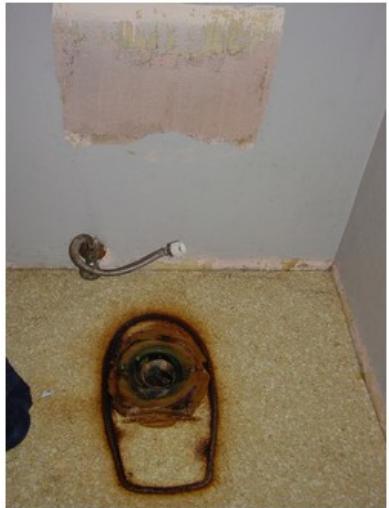


a table with a plate of food and a cup of coffee
logprob: -9.93



a young boy is playing frisbee in the park
logprob: -9.52

Not sure what happened there...



a toilet with a seat up in a bathroom
logprob: -13.44



a woman holding a teddy bear in front of a mirror
logprob: -9.65



a horse is standing in the middle of a road
logprob: -10.34

See predictions on 1000 COCO images: <http://bit.ly/neuraltalkdemo>

NeuralTalk Sentence Generation Results
Showing results for coco on 1000 images
Eval params were: ('result_struct_filename':'result_struct.json','beam_size':1,'checkpoint_path':'cv/model_checkpoint_coco_visionlab43.stanford.edu.lstm_11.14.p','dump_folder':'out1','max_images':1000)
Final average perplexity of ground truth words: 11.56

The image shows a 4x5 grid of 20 images from the COCO dataset. Each image is accompanied by a caption and a logprob value. The images include various scenes such as people walking, landscapes, food items, and animals. The captions and logprobs are as follows:

- a group of people walking down a street logprob: -5.73
- a man is standing on a beach with a surfboard logprob: -10.34
- a woman is holding a black umbrella in the grass logprob: -12.59
- a plate of food with a sandwich and a salad logprob: -9.69
- a woman sitting on a bench with a laptop logprob: -8.95
- a traffic light with a red light on top of it logprob: -10.33
- a cat laying on a bed with a laptop logprob: -10.67
- a group of people sitting at a table with wine glasses logprob: -6.71
- a bus is parked in a parking lot logprob: -8.06
- a pizza with toppings on a white plate logprob: -7.40
- a herd of sheep standing on top of a lush green field logprob: -6.22
- a giraffe standing in a field of grass logprob: -7.43
- a large clock tower with a clock on top logprob: -8.21
- a woman is holding a pink umbrella in her hand logprob: -11.47
- a plate of food with a fork and a glass of wine logprob: -11.50

What this approach Doesn't do:

- There is no *reasoning*
- A single glance is taken at the image, no objects are detected, etc.
- We can't just describe any image

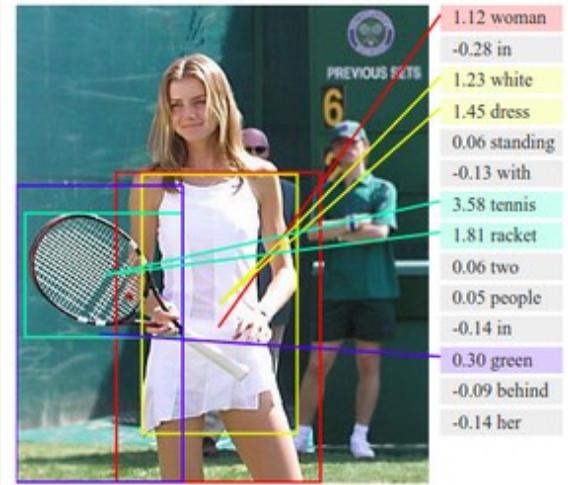
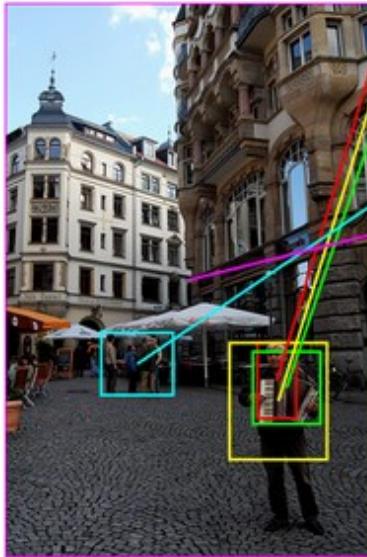
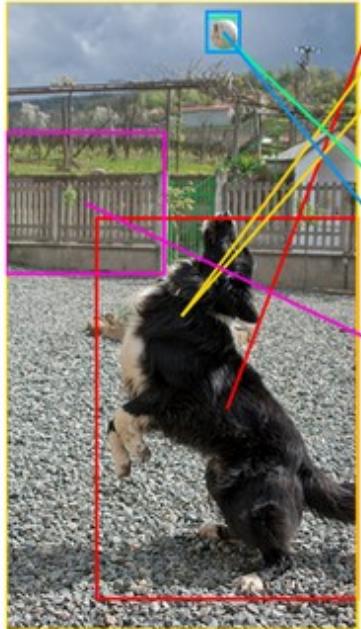
NeuralTalk

- Code on Github
- Both RNN/LSTM
- Python+numpy (CPU)
- Matlab+Caffe if you want to run on new images (for now)

The screenshot shows the GitHub repository page for 'karpathy / neuraltalk'. At the top, there are links for 'Explore', 'Gist', 'Blog', and 'Help', along with a user profile for 'karpathy'. Below the header, the repository name 'karpathy / neuraltalk' is displayed, along with a star count of 216 and a contributor count of 1. A summary bar indicates 9 commits, 1 branch, 0 releases, and 1 contributor. The 'master' branch is selected. The main content area lists 9 commits, each with a file icon, the commit message, the author ('karpathy'), the date ('9 days ago' or '2 months ago'), and a link to the commit. Below the commits is a section for 'Readme.md' containing the text: 'NeuralTalk'. At the bottom, a note states: 'This project contains Python+numpy source code for learning **Multimodal Recurrent Neural Networks** that describe images with sentences.'

File	Message	Author	Date
cv	first commit of code, phew	karpathy	2 months ago
data	first commit of code, phew	karpathy	2 months ago
eval	changing evaluation to use a perl script consistent with what the oth...	karpathy	26 days ago
example_images	adding ability to predict on arbitrary images and the script that run...	karpathy	9 days ago
imagernn	adding ability to feed image only once in beginning. works better.	karpathy	26 days ago
matlab_features_reference	first commit of code, phew	karpathy	2 months ago
status	first commit of code, phew	karpathy	2 months ago
vis_resources	first commit of code, phew	karpathy	2 months ago
.gitignore	first commit of code, phew	karpathy	2 months ago
Readme.md	adding ability to predict on arbitrary images and the script that run...	karpathy	9 days ago
driver.py	adding ability to feed image only once in beginning. works better.	karpathy	26 days ago
eval_sentence_predictions.py	adding ability to dump images that are evaluated on into some folder	karpathy	9 days ago
monitorcv.html	first commit of code, phew	karpathy	2 months ago
predict_on_images.py	adding ability to predict on arbitrary images and the script that run...	karpathy	9 days ago
visualize_result_struct.html	simplifying. no need for svg element here	karpathy	26 days ago

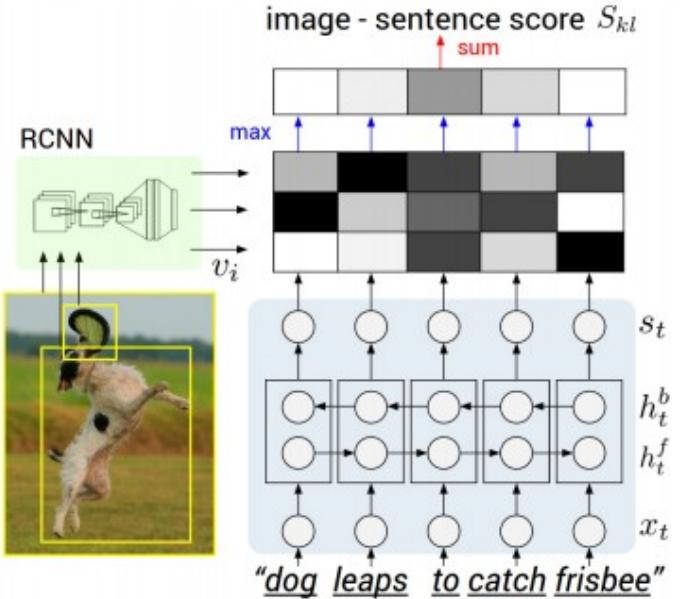
Ranking model



Ranking model

web demo:

<http://bit.ly/rankingdemo>



Grounded Image Sentence Retrieval

For every test set sentence below we retrieve the top images (from set of 1000). Yellow number top left of each image = score. Clicking on each image reveals the precise inferred grounding. Red border = incorrect retrieval, green border = correct retrieval. Yellow border = ground truth image that wasn't retrieved among top 5 predictions.

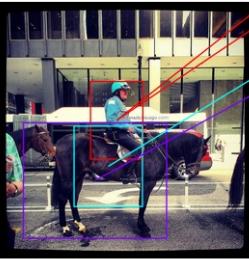
Test set of size: 1000 images

regex filter sentences

A cop riding on the back of a black horse.

3.29  3.16  2.91  2.36  2.30 

0.96 cop
0.57 riding
-0.35 on
-0.17 back
0.59 black
1.56 horse



top sentences for this image:

(4.82) a horse cop sitting on a horse on a public street.
 (3.65) The drawing depicts a man with a horse and two people on the porch.
 (3.60) An older man is riding a horse down the street.
 (3.59) A man riding a horse in front of a fence.
 (3.56) A man on a horse walking down the middle of a street.

The cat is laying down while someone rubs its head

2.13  2.78  2.55  2.31  2.31 

The man in the dark suit stands in the dark room.

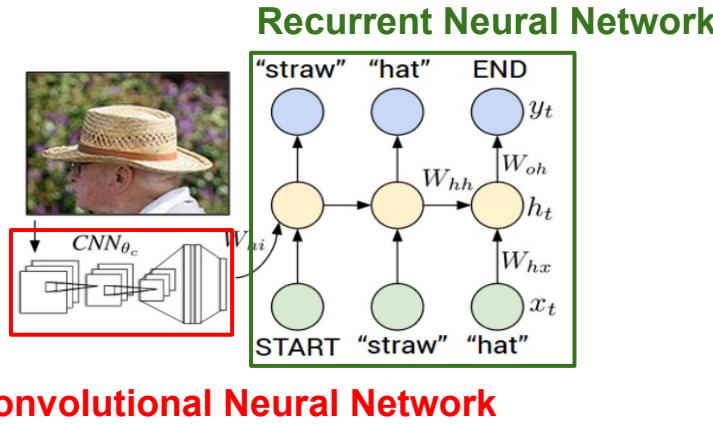
2.03  1.73  1.36  1.31  1.17 

A very large farmers market with customers waiting for samples.

Summary

Neural Networks:

- input->output **end-to-end** optimization
- stackable / composable like Lego
- easily support Transfer Learning
- work very well.



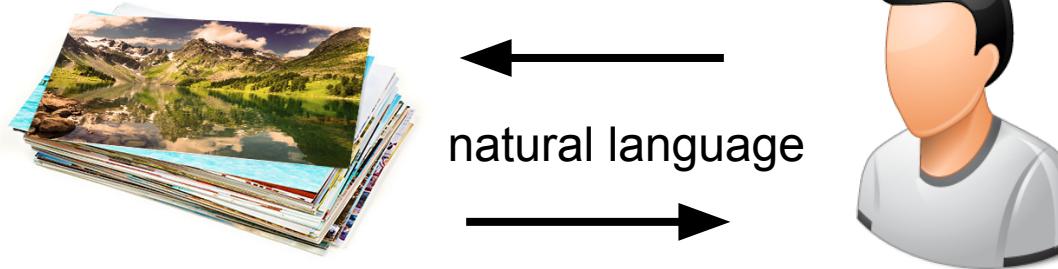
Convolutional Neural Network

Summary

1. image -> sentence
2. sentence -> image

Summary

1. image -> sentence
2. sentence -> image



Summary

1. image -> sentence
2. sentence -> image



natural language



Thank you!