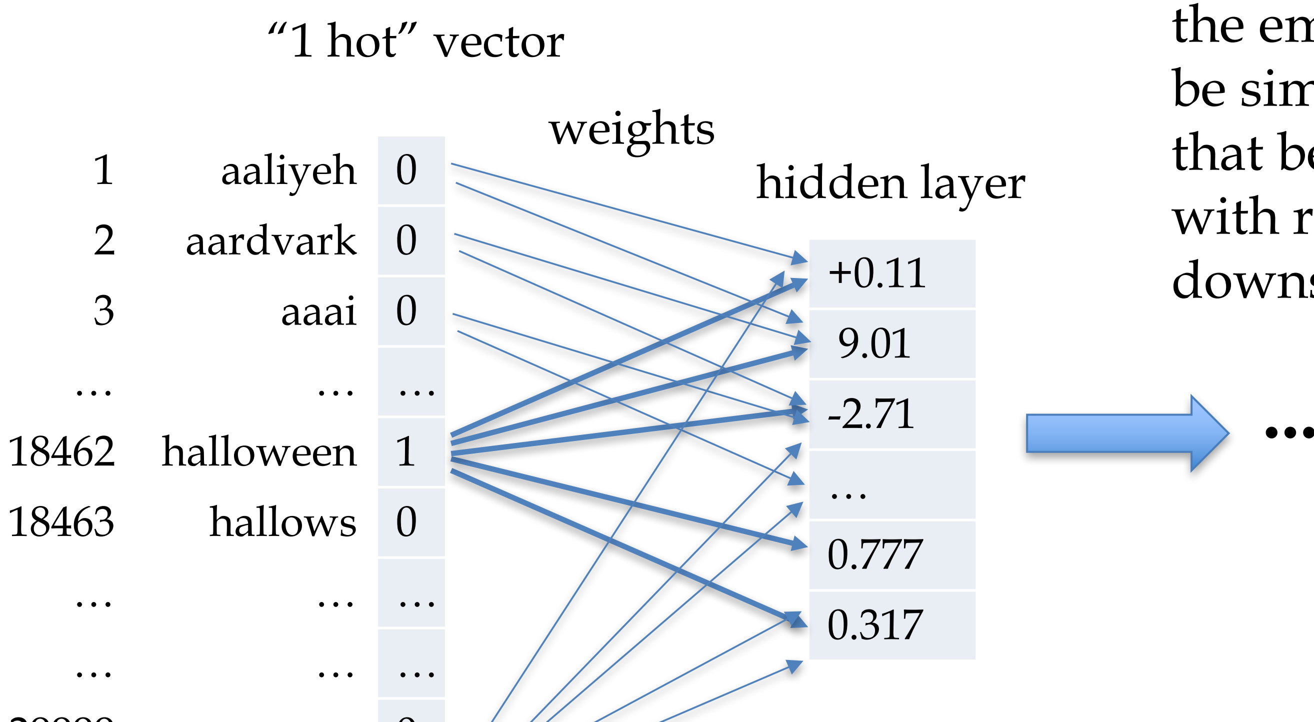

WORLD'S FIRST NON-CONTACT
VITAL BABY MONITOR
UPTO 30% OFF*
It's
Halloween
AND WE HAVE
A TREAT
OFFER LAST TILL 31ST OCTOBER
raybaby™
SUPPORTED BY
JOHNSON & JOHNSON AND HAX
as part of the Consumer Health Device Program

Deep Architectures

Word2Vec and GloVe Embeddings

Representing words in a deep network



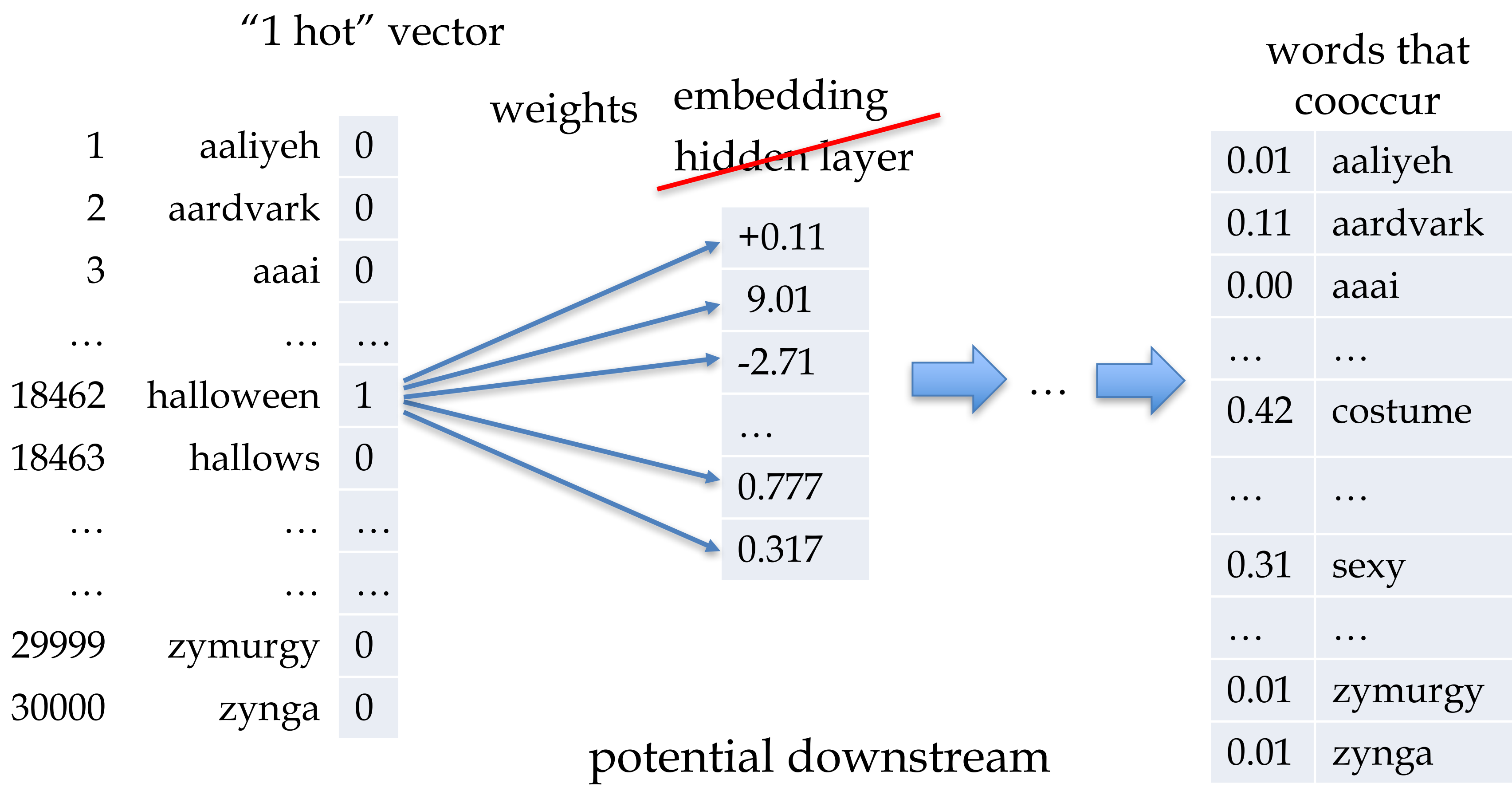
the embeddings will be similar for words that behave similarly with respect to the downstream task

$\mathbf{h} = \mathbf{x}\,W$

but really **h** is the i-th row of W

so learning W is just learning a hidden-layer encoding for each word in the vocabulary (**embedding**)

Representing words in a deep network

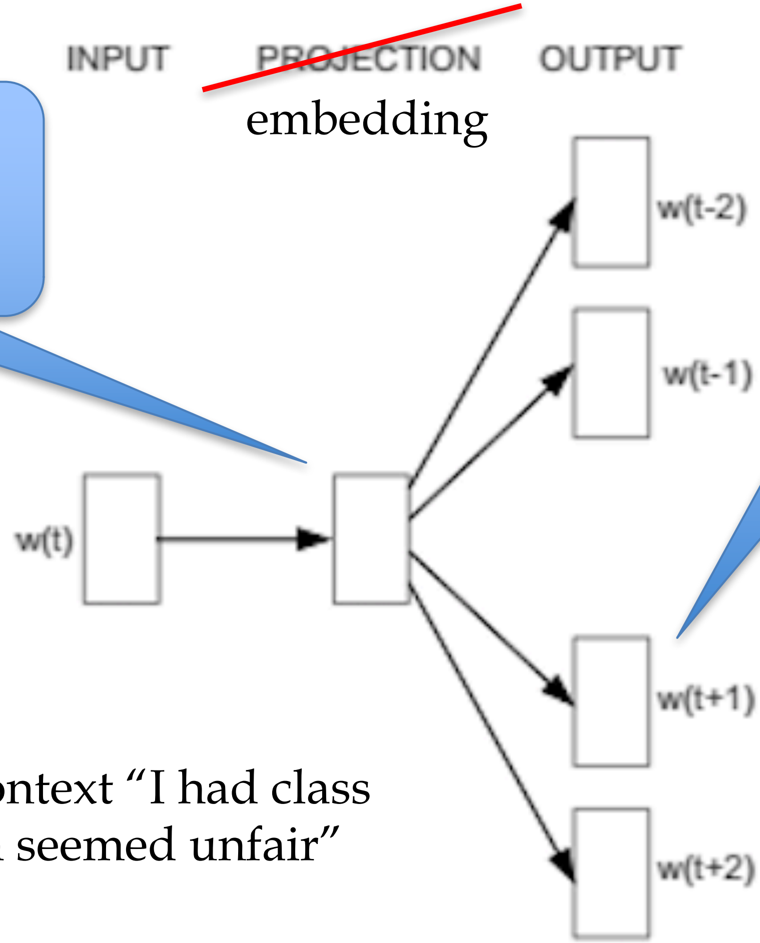


potential downstream task: predict the words that co-occur with input word

word2vec: skip-gram embeddings

vector that "encodes" the t-th word

from an input word w(t) in a document



Skip-gram

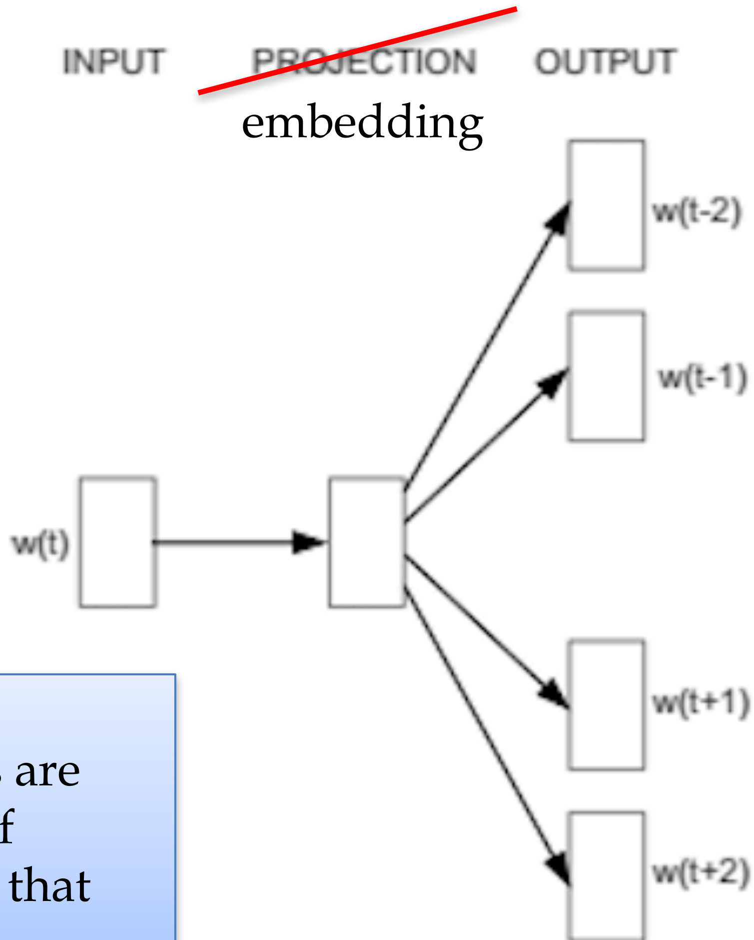
So that the hidden layer will predict likely nearby words w(t-K), …, w(t+K)

final step of this prediction is a softmax over the vocabulary

A word in context context "I had class on Halloween which seemed unfair" becomes an example

word2vec: skip-gram embeddings

Training data: **positive** examples are pairs of words w(t), w(t+j) that co-occur



Skip-gram

You want to train over a very large corpus (100M words+) and hundreds+ dimensions

Training data: negative examples are **samples** of pairs of words w(t), w(t+j) that don't co-occur

GLOVE embeddings

$$f(x) = \begin{cases} (x/x_{\max})^{\alpha} & \text{if } x < x_{\max} \\ 1 & \text{otherwise} \end{cases}$$

how often words *i* and *j* co-occur in a corpus

$$J = \sum_{i,j=1}^{V} f\left(X_{ij}\right)\left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2$$

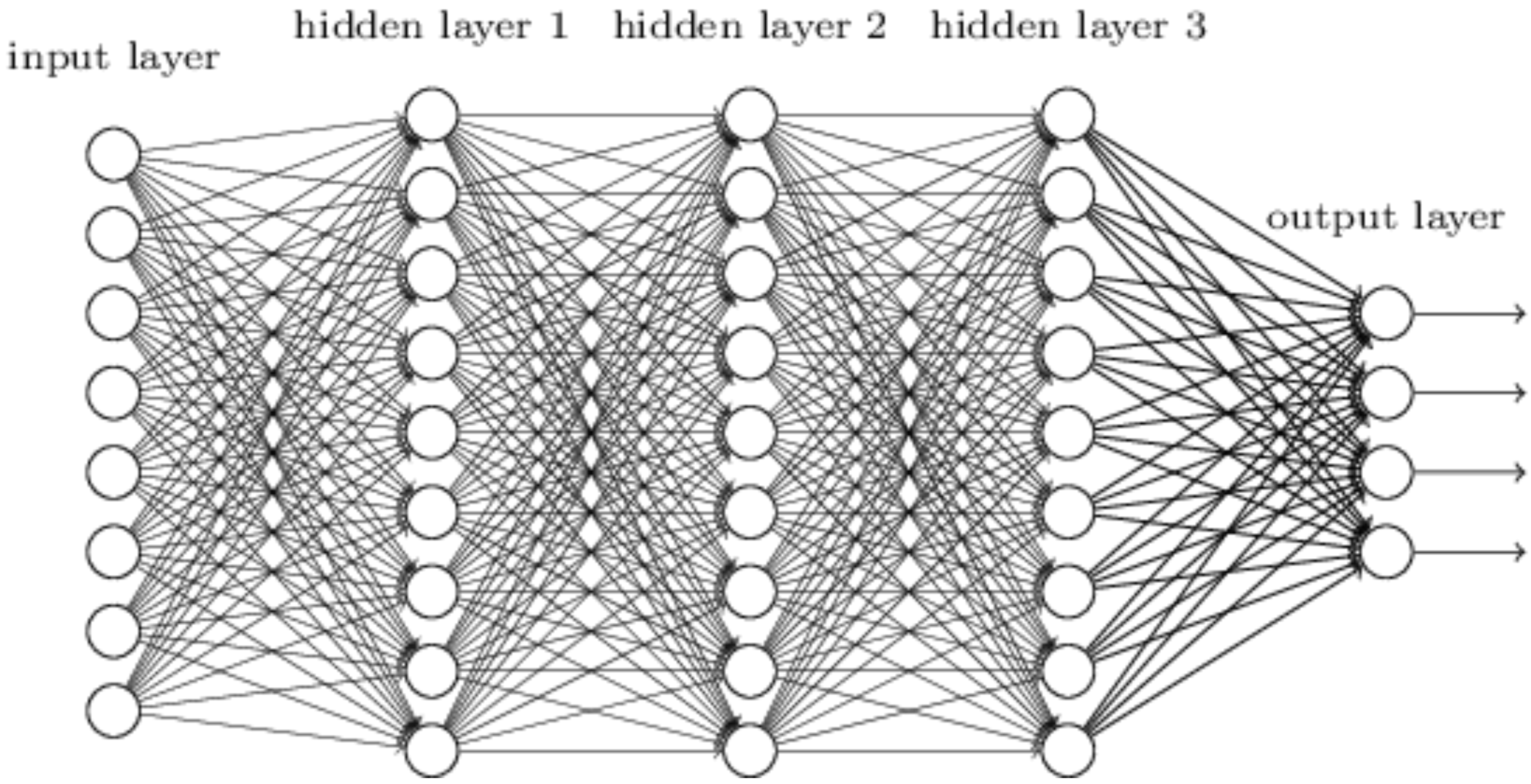
how much to weight this word pair, based on frequency

embeddings for words *i* and *j*

biases for words *i* and *j*

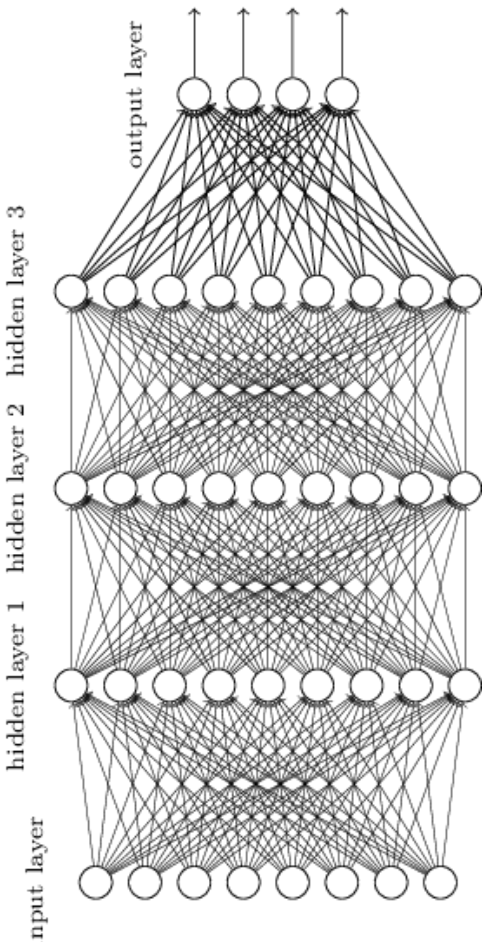
RECURRENT NEURAL NETWORKS

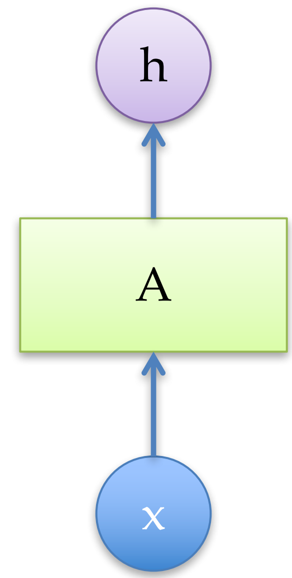
Motivation: what about sequence prediction?



What can I do when input size and output size vary?

Motivation: what about sequence prediction?

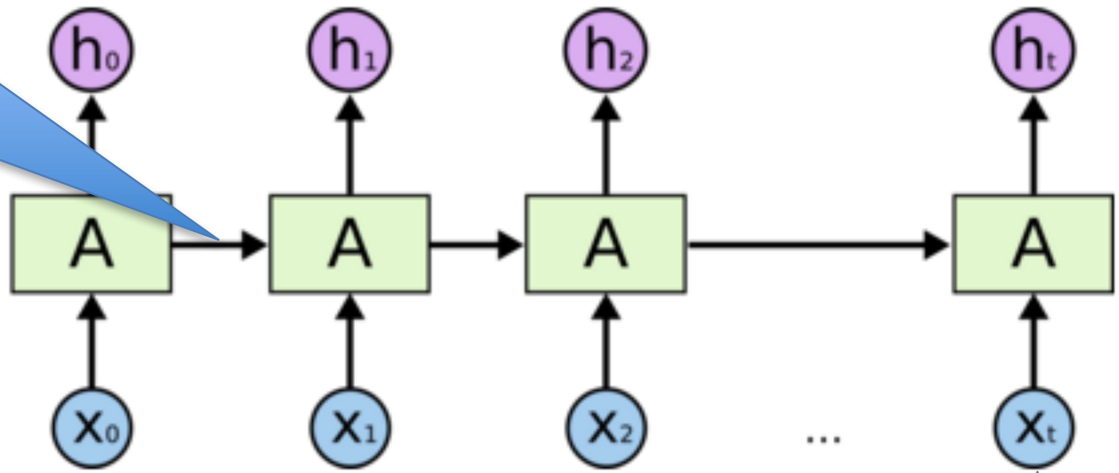




Architecture for an RNN

Some information is passed from one subunit to the next

Sequence of outputs



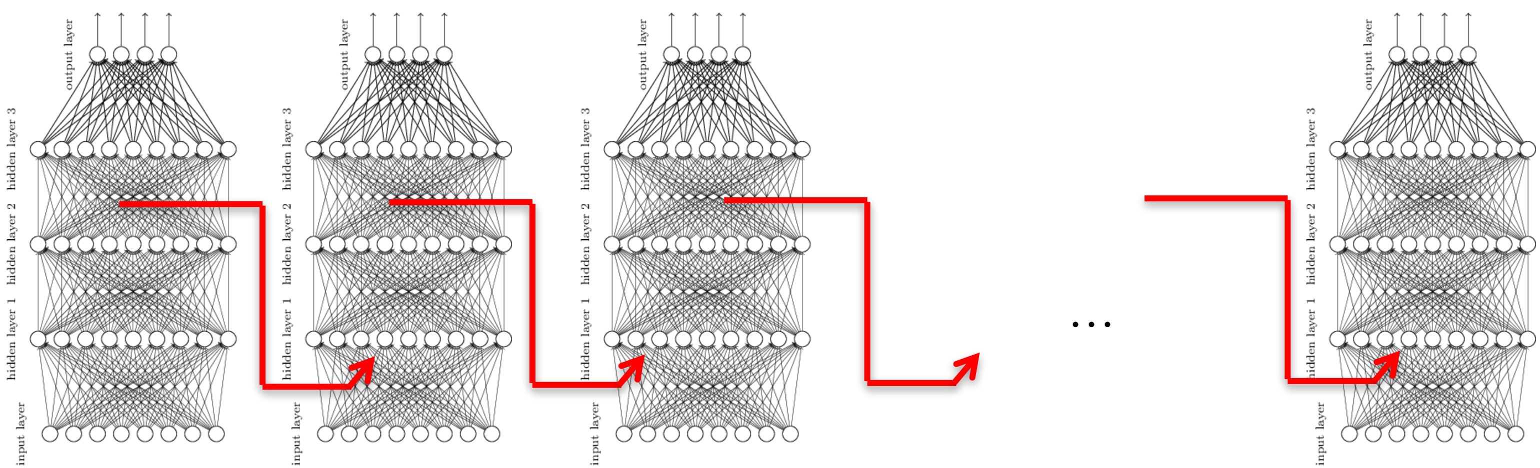
Start of sequence marker

Sequence of inputs

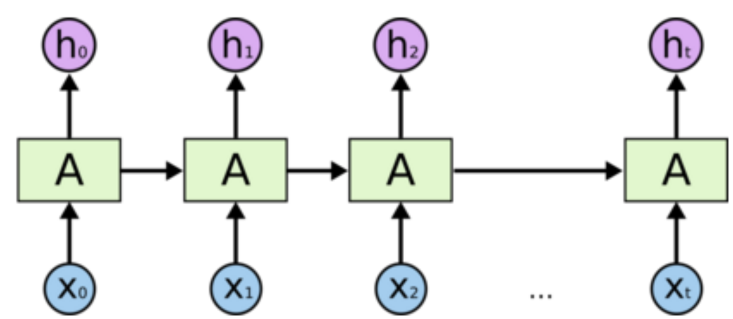
End of sequence marker

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

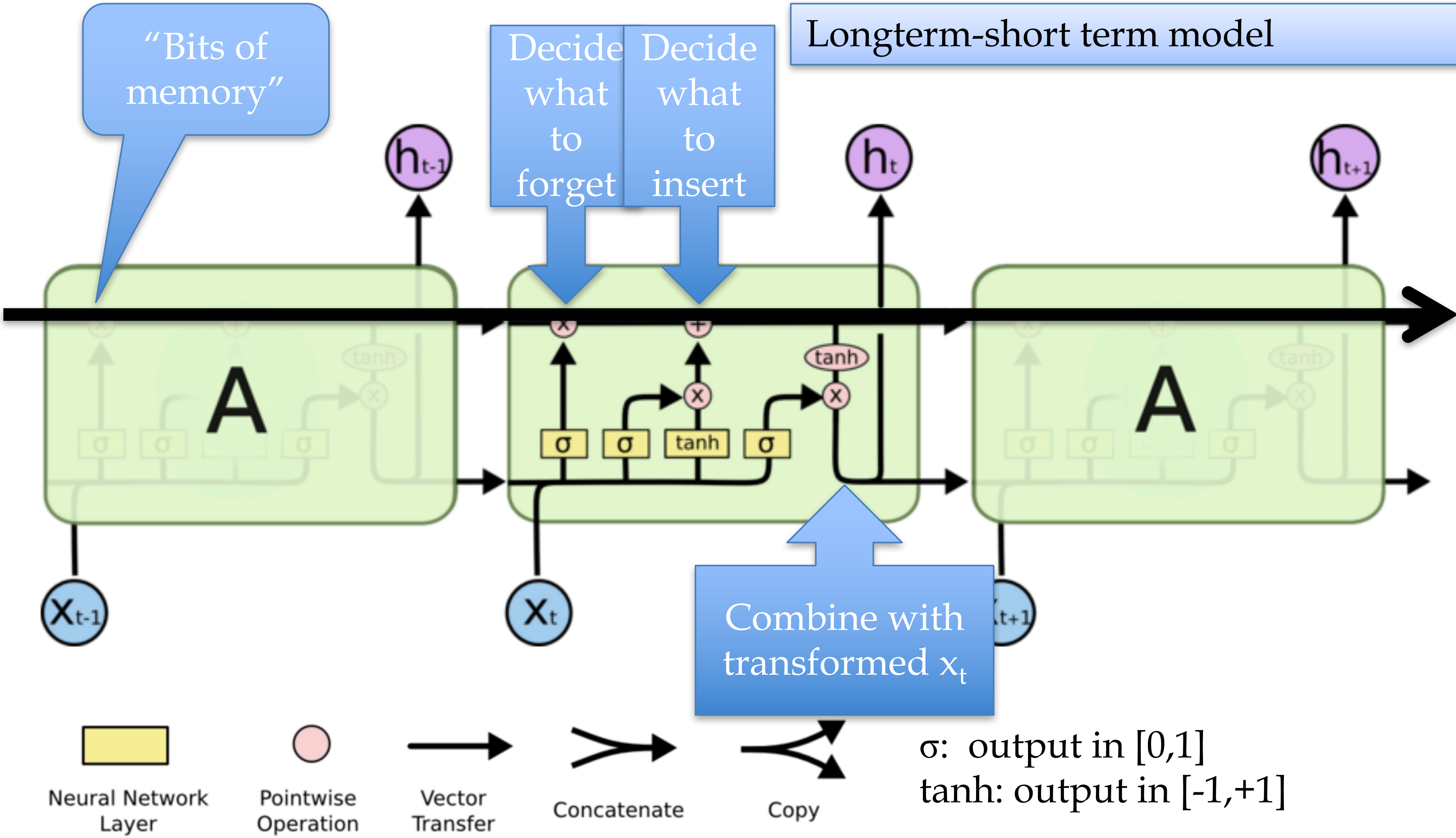
Architecture for an 1980's RNN



Problem with this: it's extremely deep and very hard to train



Architecture for an LSTM

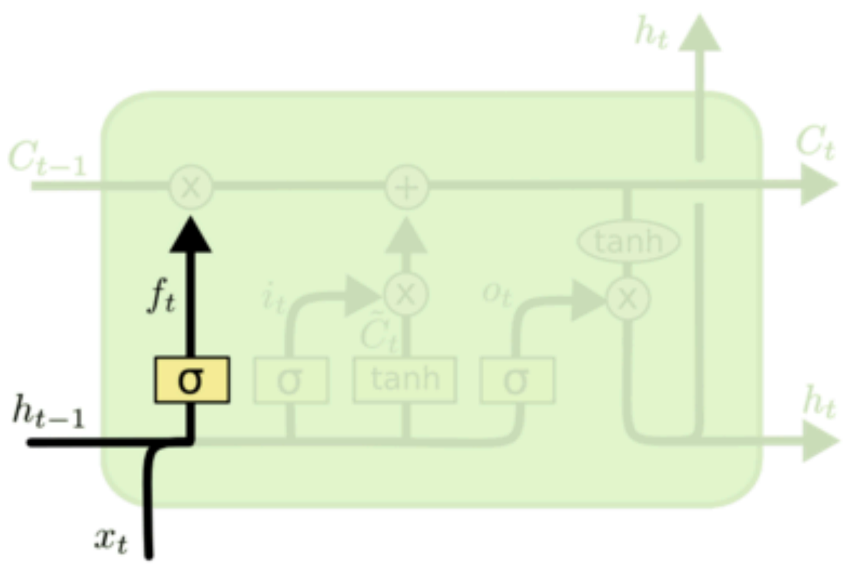


σ: output in [0,1]

tanh: output in [-1,+1]

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

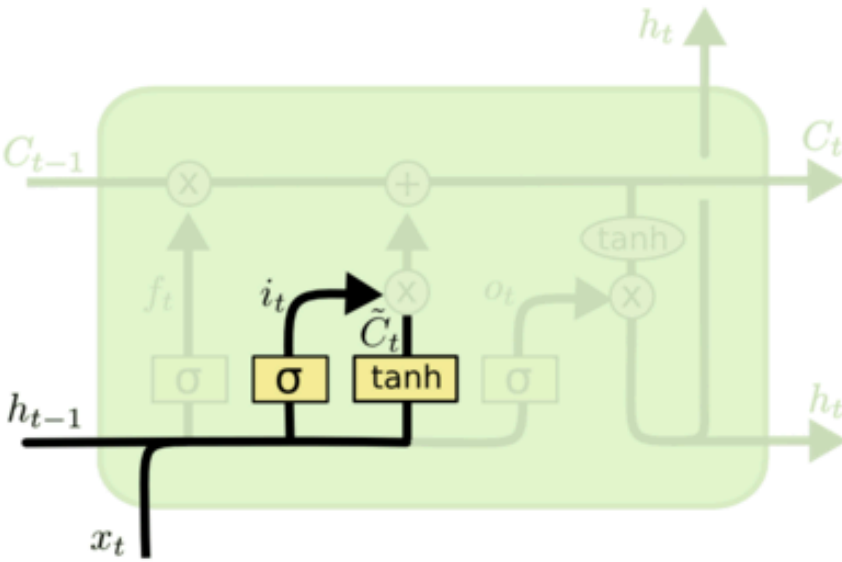
Walkthrough



What part of memory to "forget" – zero means forget this bit

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

Walkthrough



What bits to insert into the next states

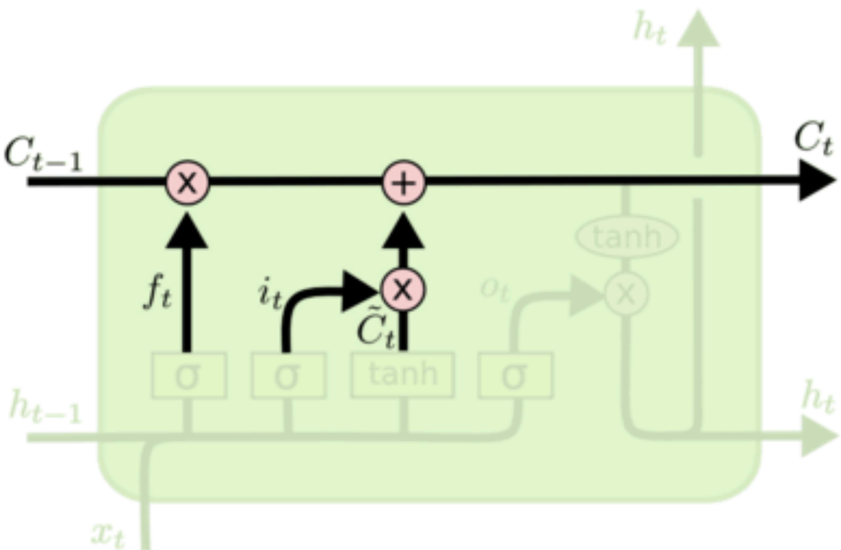
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$

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

What content to store into the next state

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

Walkthrough



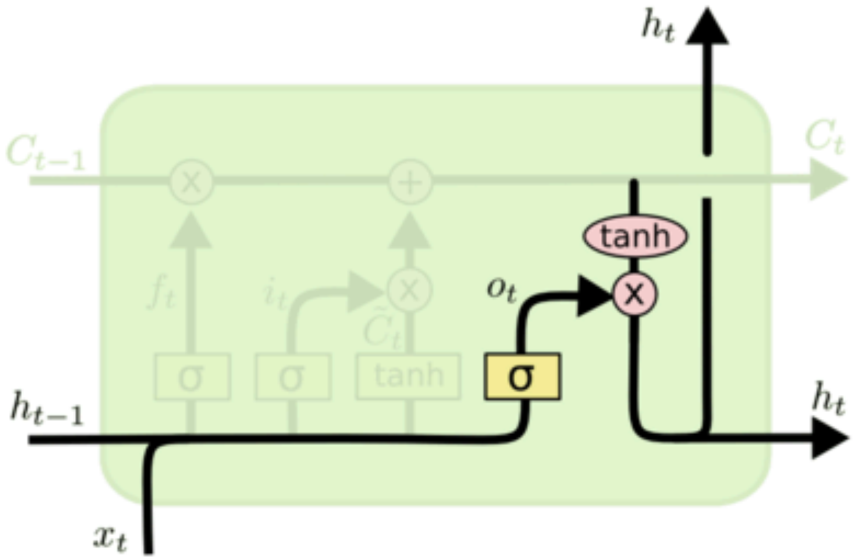
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Next memory cell content – mixture of not-forgotten part of previous cell and insertion

This is the important part! the LSTM can pass data throuh unchanged

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

Walkthrough



What part of cell to output

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

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

tanh maps bits to [-1,+1] range

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

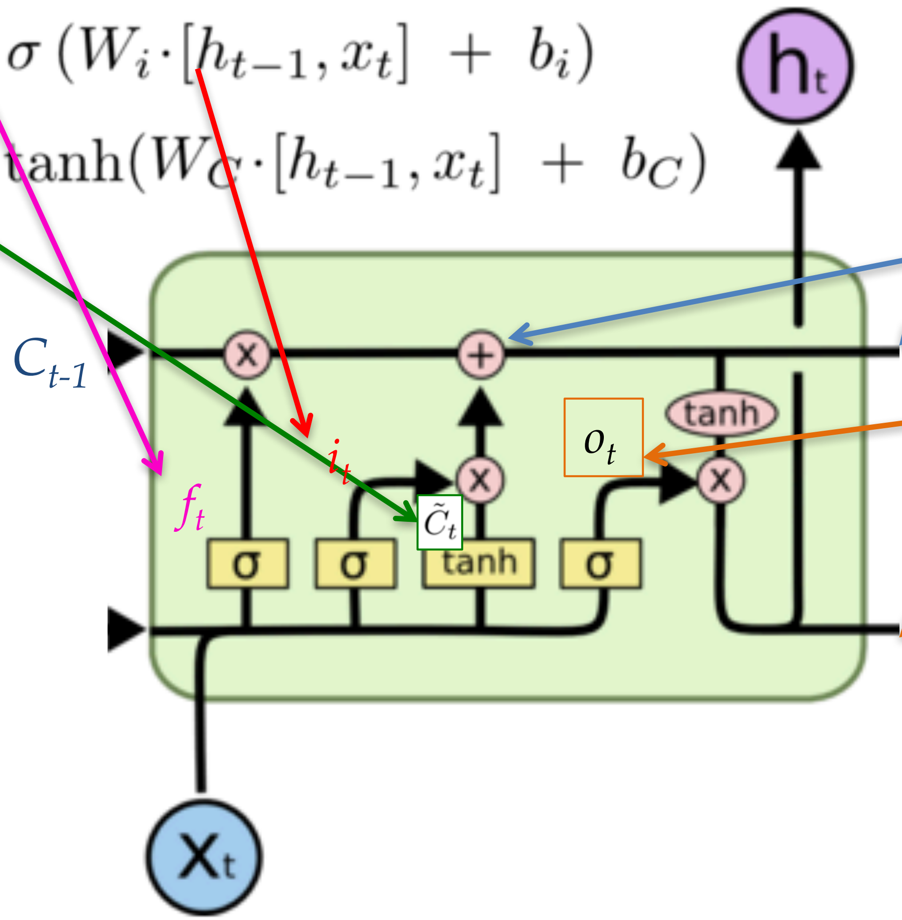
Architecture for an LSTM

(1)

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

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

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



(2)

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

(3)

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

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

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

Implementing an LSTM

For $t = 1,\ldots,T$:

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

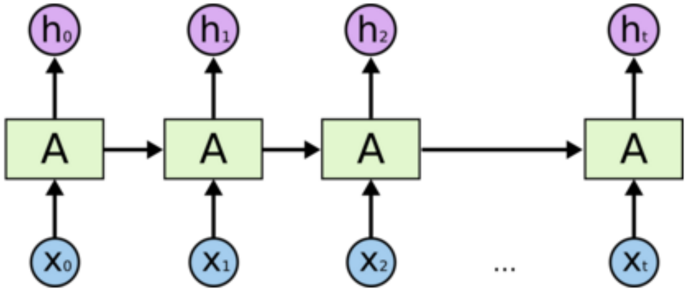
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$

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

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

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

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

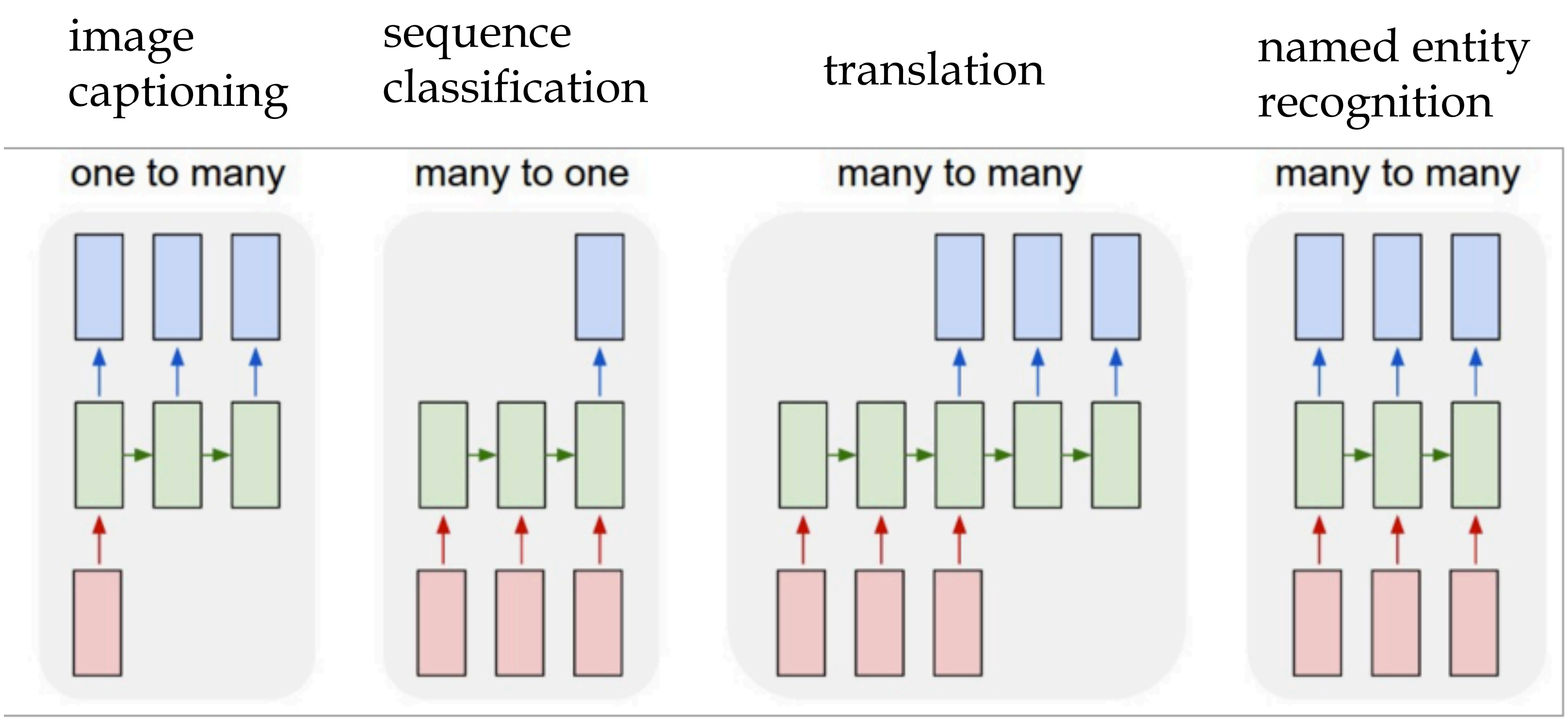


http://colah.github.io/posts/2015-08-Understanding-LSTMs/

SOME FUN LSTM EXAMPLES

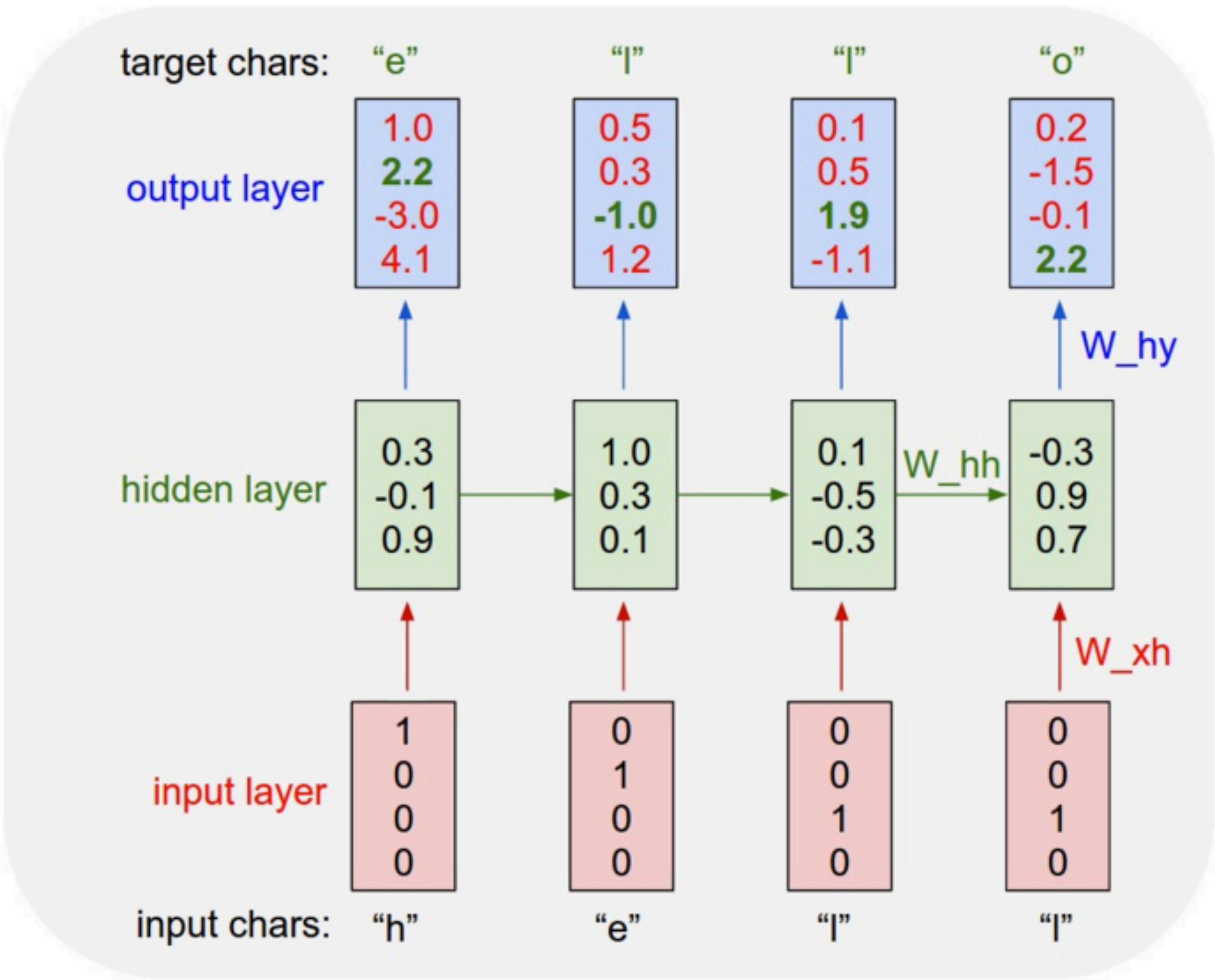
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

LSTMs can be used for other sequence tasks



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

Character-level language model



Test time:

- pick a seed character sequence
- generate the next character
- then the next
- then the next …

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

Character-level language model

```
PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.
```

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

Character-level language model

```
First Citizen:
Nay, then, that was hers,
It speaks against your other service:
But since the
youth of the circumstance be spoken:
Your uncle and one Baptista's daughter.

SEBASTIAN:
Do I stand till the break off.

BIRON:
Hide thy head.

VENTIDIUS:
He purposeth to Athens: whither, with the vow
I made to handle you.

FALSTAFF:
My good knave.
```

Yoav Goldberg: order-10 unsmoothed character n-grams

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

Character-level language model

```
MMMMM----- Recipe via Meal-Master (tm) v8.05

      Title: BARBECUE RIBS
 Categories: Chinese, Appetizers
      Yield: 4 Servings

      1 pk Seasoned rice
      1    Beer -- cut into
           -cubes
      1 ts Sugar
    3/4 c  Water
           Chopped finels,
           -up to 4 tblsp of chopped
      2 pk Yeast Bread/over

MMMMM--------------------------FILLING------------------------------
      2 c  Pineapple, chopped
    1/3 c  Milk
    1/2 c  Pecans
           Cream of each
      2 tb Balsamic cocoa
      2 tb Flour
      2 ts Lemon juice
           Granulated sugar
      2 tb Orange juice
```

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

Character-level language model

For $\bigoplus_{n=1,\ldots,m}$ where $\mathcal{L}_{m_\bullet} = 0$, hence we can find a closed subset $\mathcal{H}$ in $\mathcal{H}$ and any sets $\mathcal{F}$ on X, U is a closed immersion of S, then $U \to T$ is a separated algebraic space.

Proof. Proof of (1). It also start we get

$$S = \mathrm{Spec}(R) = U \times_X U \times_X U$$

and the comparicoly in the fibre product covering we have to prove the lemma generated by $\coprod Z \times_U U \to V$. Consider the maps M along the set of points Sch_{fppf} and $U \to U$ is the fibre category of S in U in Section, **??** and the fact that any U affine, see Morphisms, Lemma **??**. Hence we obtain a scheme S and any open subset $W \subset U$ in $Sh(G)$ such that $\mathrm{Spec}(R') \to S$ is smooth or an

$$U = \bigcup U_i \times_{S_i} U_i$$

which has a nonzero morphism we may assume that f_i is of finite presentation over S. We claim that $\mathcal{O}_{X,x}$ is a scheme where $x, x', s'' \in S'$ such that $\mathcal{O}_{X,x'} \to \mathcal{O}'_{X',x'}$ is separated. By Algebra, Lemma **??** we can define a map of complexes $\mathrm{GL}_{S'}(x'/S'')$ and we win. $\square$

To prove study we see that $\mathcal{F}|_U$ is a covering of $\mathcal{X}'$, and $\mathcal{T}_i$ is an object of $\mathcal{F}_{X/S}$ for $i > 0$ and $\mathcal{F}_p$ exists and let $\mathcal{F}_i$ be a presheaf of $\mathcal{O}_X$-modules on $\mathcal{C}$ as a $\mathcal{F}$-module. In particular $\mathcal{F} = U/\mathcal{F}$ we have to show that

$$\widetilde{M}^\bullet = \mathcal{I}^\bullet \otimes_{\mathrm{Spec}(k)} \mathcal{O}_{S,s} - i_X^{-1}\mathcal{F})$$

is a unique morphism of algebraic stacks. Note that

$$\mathrm{Arrows} = (Sch/S)^{opp}_{fppf}, (Sch/S)_{fppf}$$

and

$$V = \Gamma(S, \mathcal{O}) \longmapsto (U, \mathrm{Spec}(A))$$

LaTeX "almost compiles"

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

Character-level language model

```
/*
 * Increment the size file of the new incorrect UI_FILTER group information
 * of the size generatively.
 */
static int indicate_policy(void)
{
  int error;
  if (fd == MARN_EPT) {
    /*
     * The kernel blank will coeld it to userspace.
     */
    if (ss->segment < mem_total)
      unblock_graph_and_set_blocked();
    else
      ret = 1;
    goto bail;
  }
  segaddr = in_SB(in.addr);
  selector = seg / 16;
  setup_works = true;
  for (i = 0; i < blocks; i++) {
    seq = buf[i++];
    bpf = bd->bd.next + i * search;
    if (fd) {
      current = blocked;
    }
  }
```

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

More examples

https://medium.com/aifromscratch/when-janelle-shane-trains-rnns-dcd4c3fa9d3d

- bleedwood 187 191 172
- parp green 110 117 72
- peacake bring 229 206 186
- flipper 159 179 186
- lemon nose 236 203 161
- shy bather 187 198 197
- spiced rope 85 90 79
- polar forest ma pepper 170 16
- windled waters 186 206 229
- barkying white 243 231 206
- clay cow 161 193 172
- dry custard 225 175 134

Volkswagen Colon

Buick Shoat

Buick Crapara

Buick Apron

Fiat Deter

Fiat Coma

Fiat S-0-S

Fiat Doug

Facial Agoricosis

Strecting Dissection of the Breath

Bacterial Fradular Syndrome

Loss Of Consufficiency

Hemopheritis

Joint Pseudomalabia

Hammon Expressive Foot

Clob

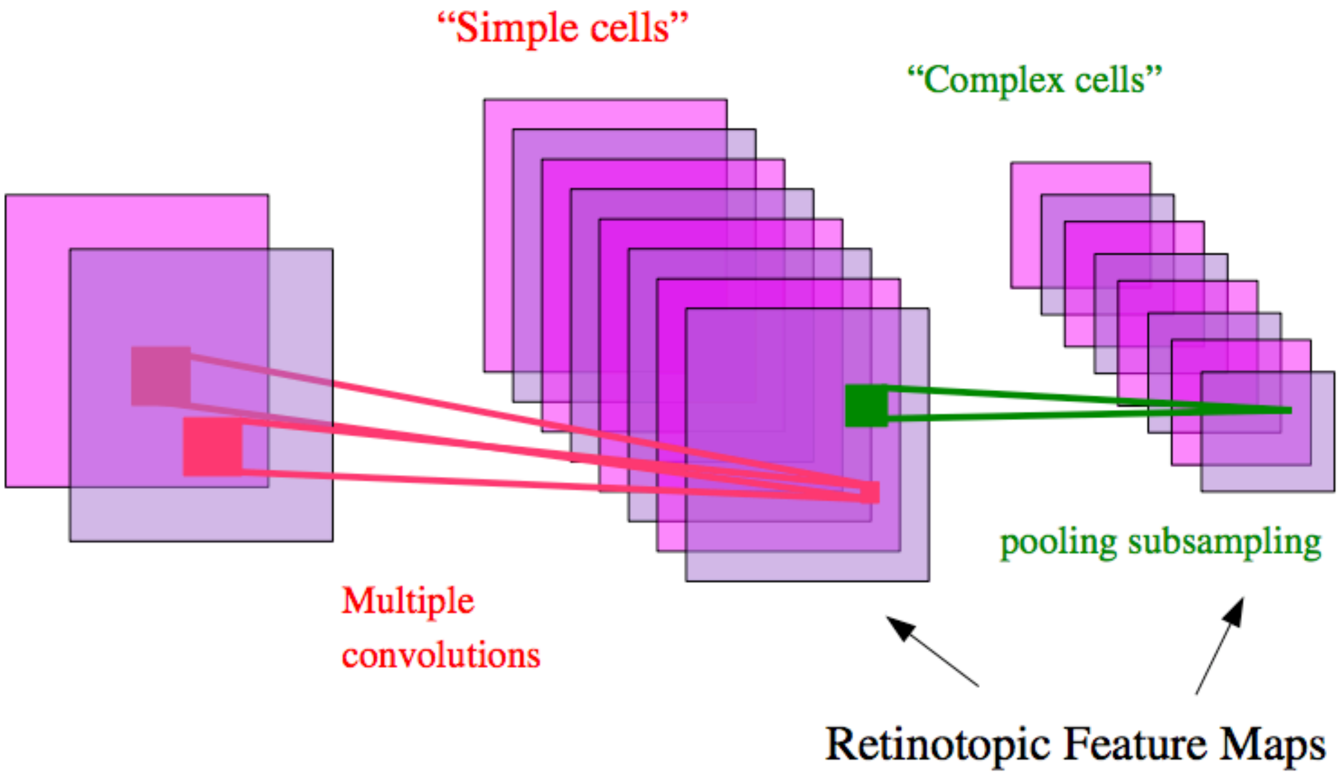
Cancer of the Cancer

Horse Stools

CONVOLUTIONAL NEURAL NETWORKS

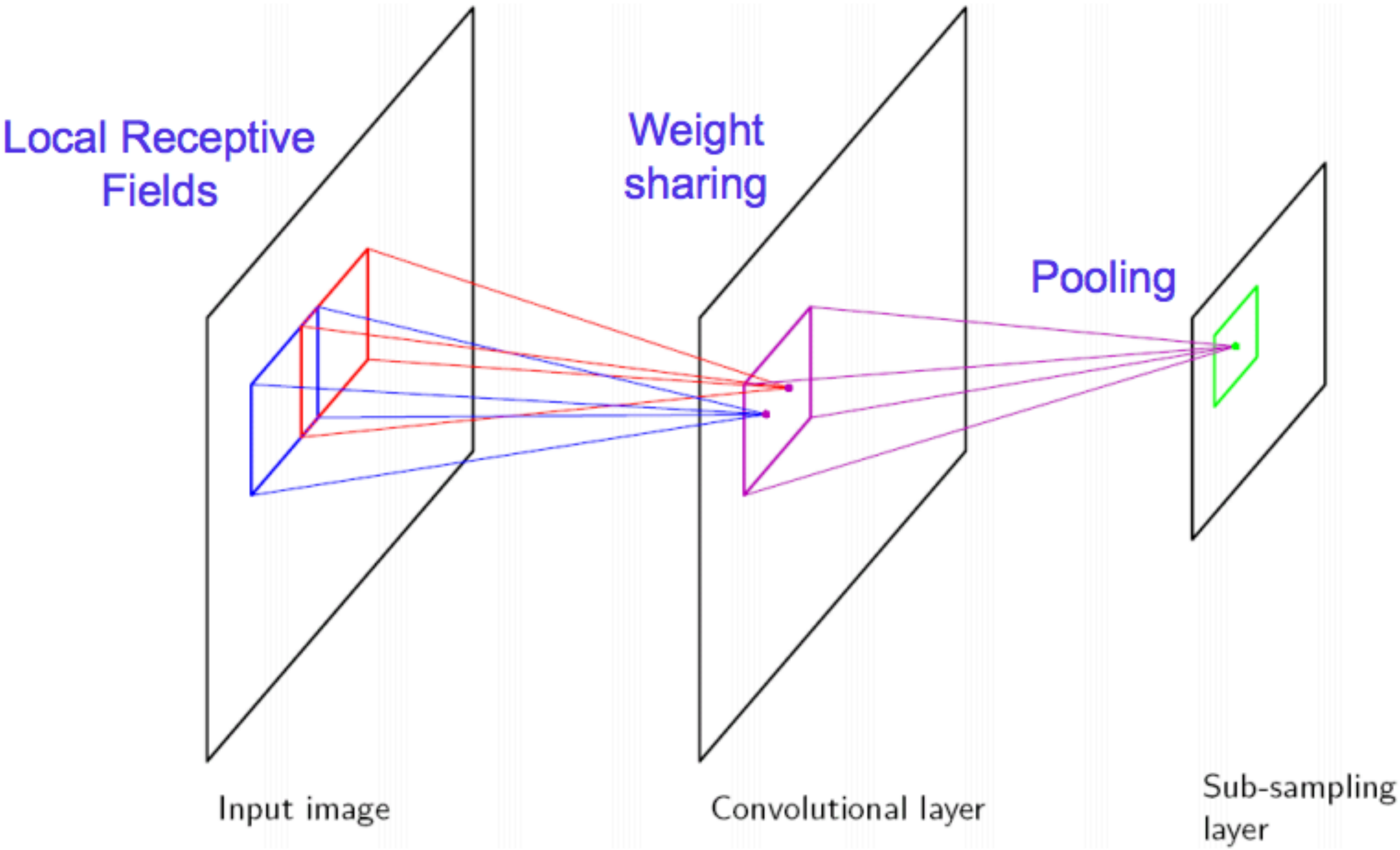
Model of vision in animals

- **[Hubel & Wiesel 1962]:**
  - simple cells detect local features
  - complex cells "pool" the outputs of simple cells within a retinotopic neighborhood.



Vision with ANNs

(LeCun et al., 1989)

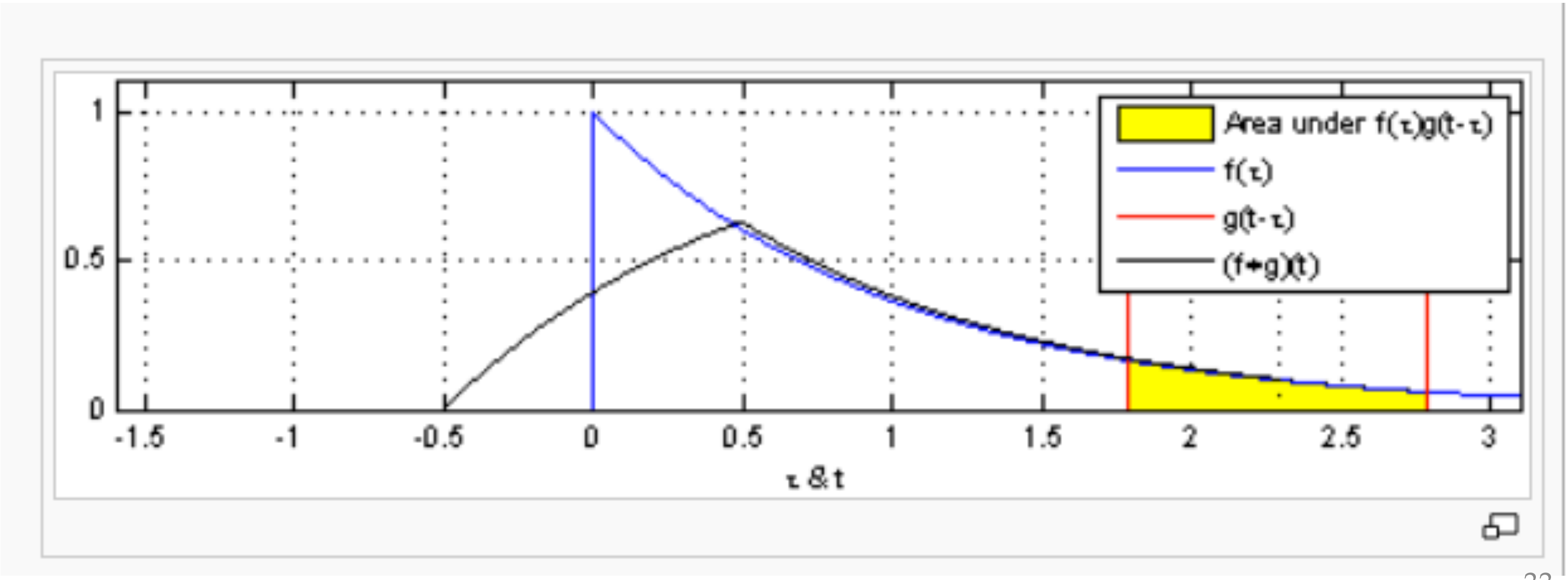


What's a convolution?

https://en.wikipedia.org/wiki/Convolution

1-D

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$$
$$= \int_{-\infty}^{\infty} f(t-\tau)\, g(\tau)\, d\tau.$$

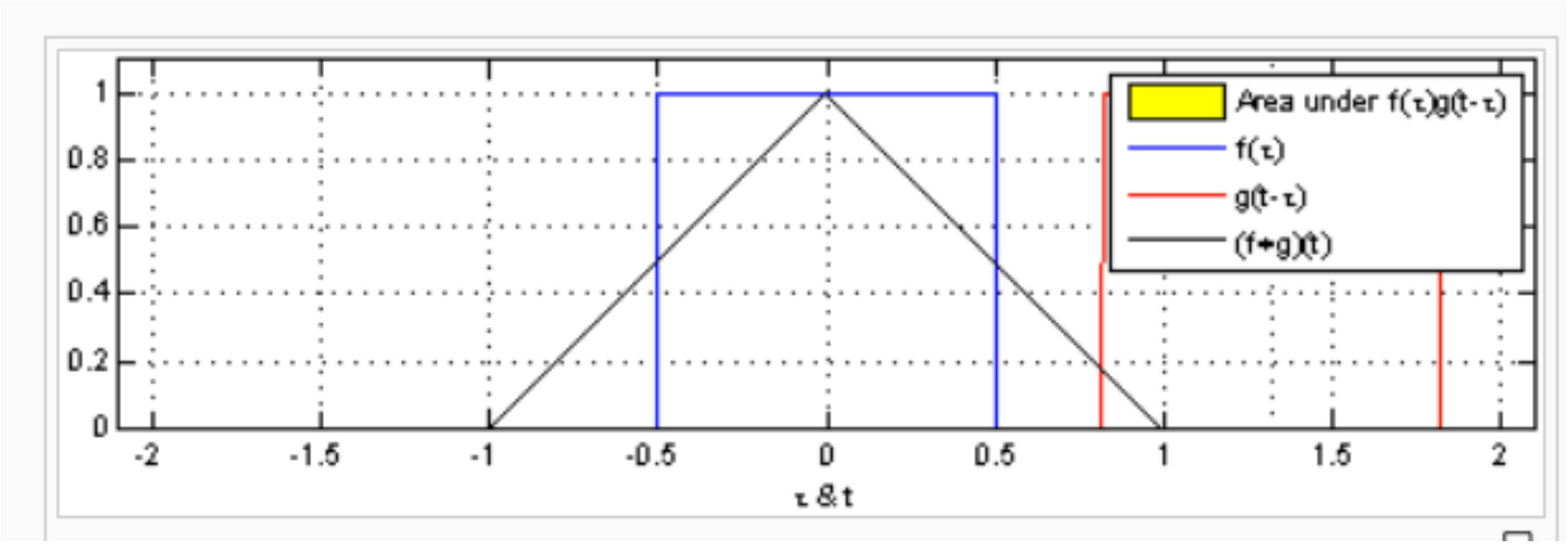


What's a convolution?

https://en.wikipedia.org/wiki/Convolution

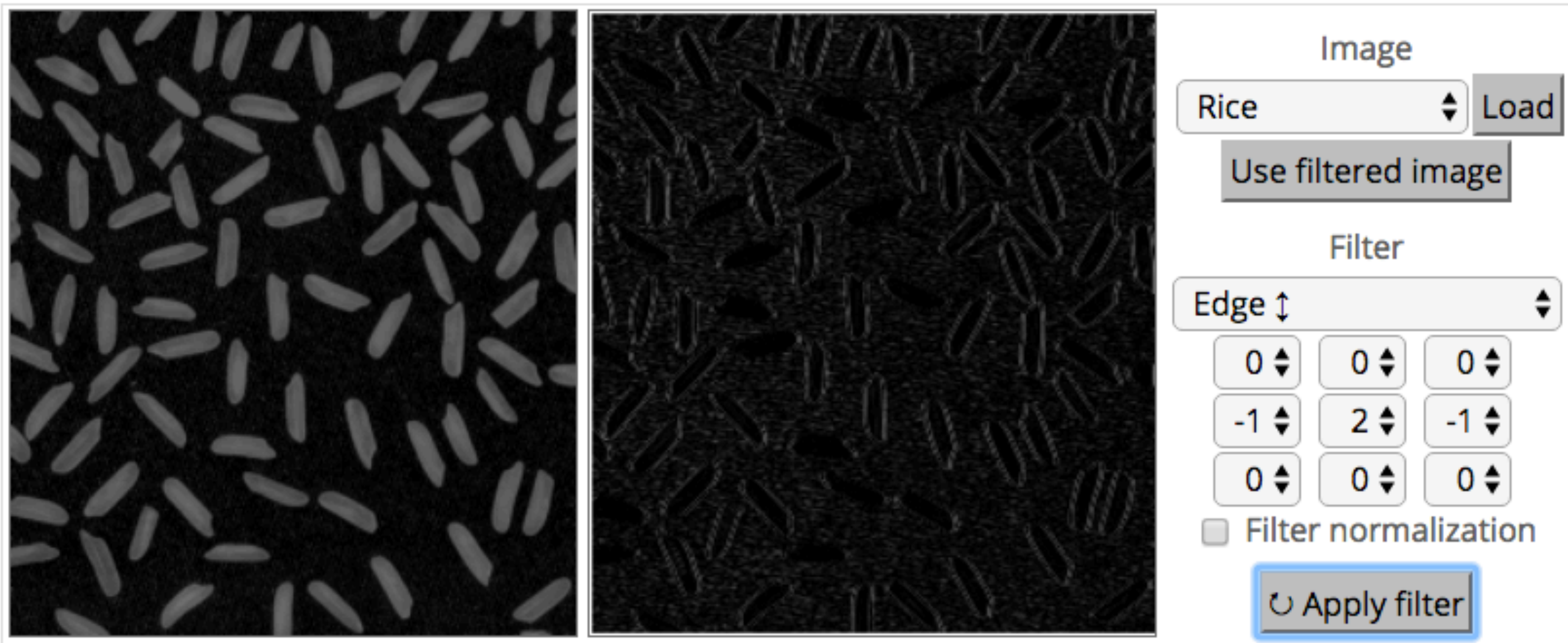
1-D

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$$
$$= \int_{-\infty}^{\infty} f(t-\tau)\, g(\tau)\, d\tau.$$



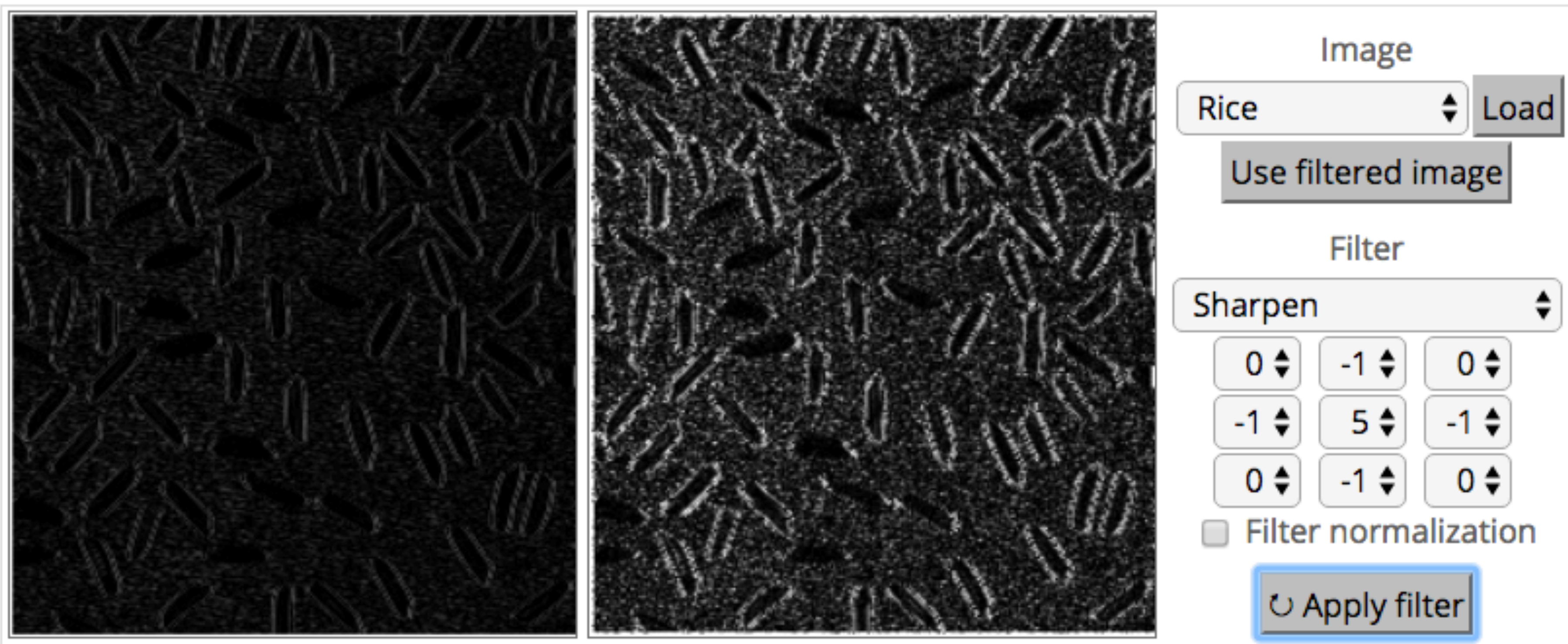
What’s a convolution?

http://matlabtricks.com/post-5/3x3-convolution-kernels-with-online-demo



What's a convolution?

http://matlabtricks.com/post-5/3x3-convolution-kernels-with-online-demo

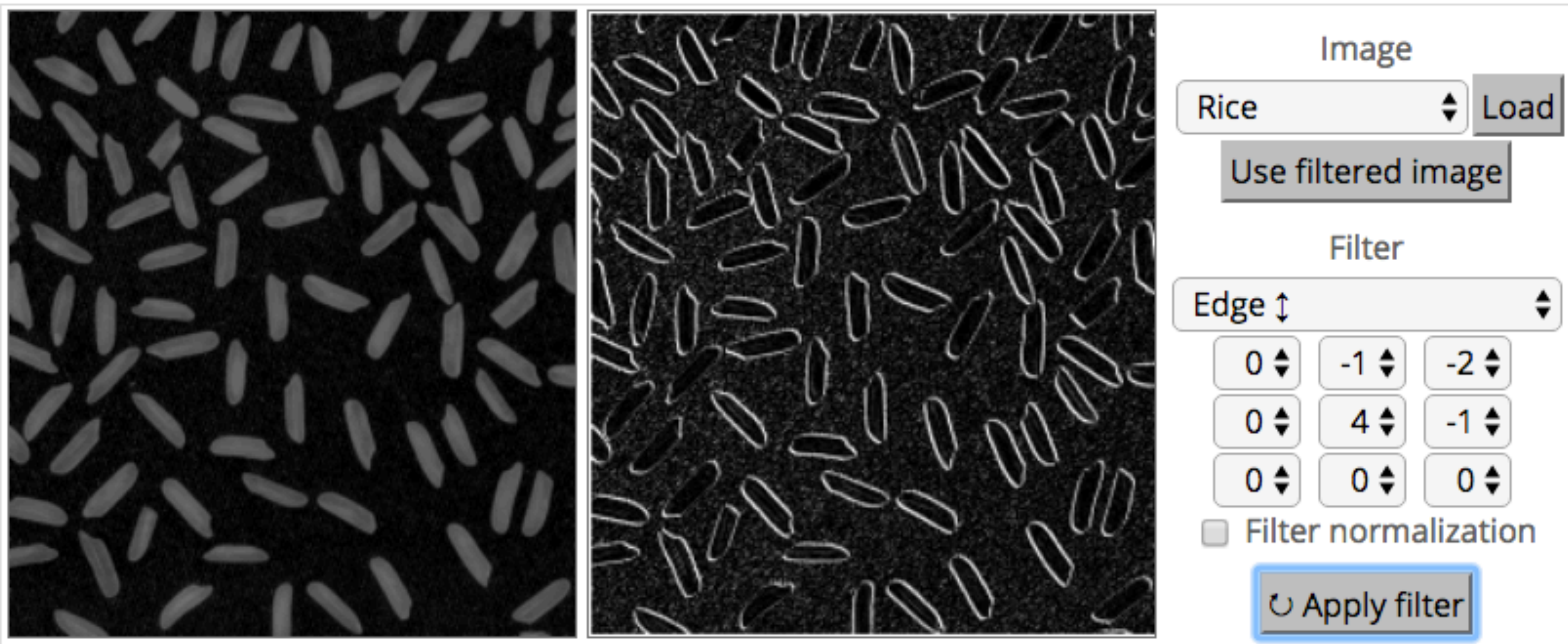


What's a convolution?

http://matlabtricks.com/post-5/3x3-convolution-kernels-with-online-demo

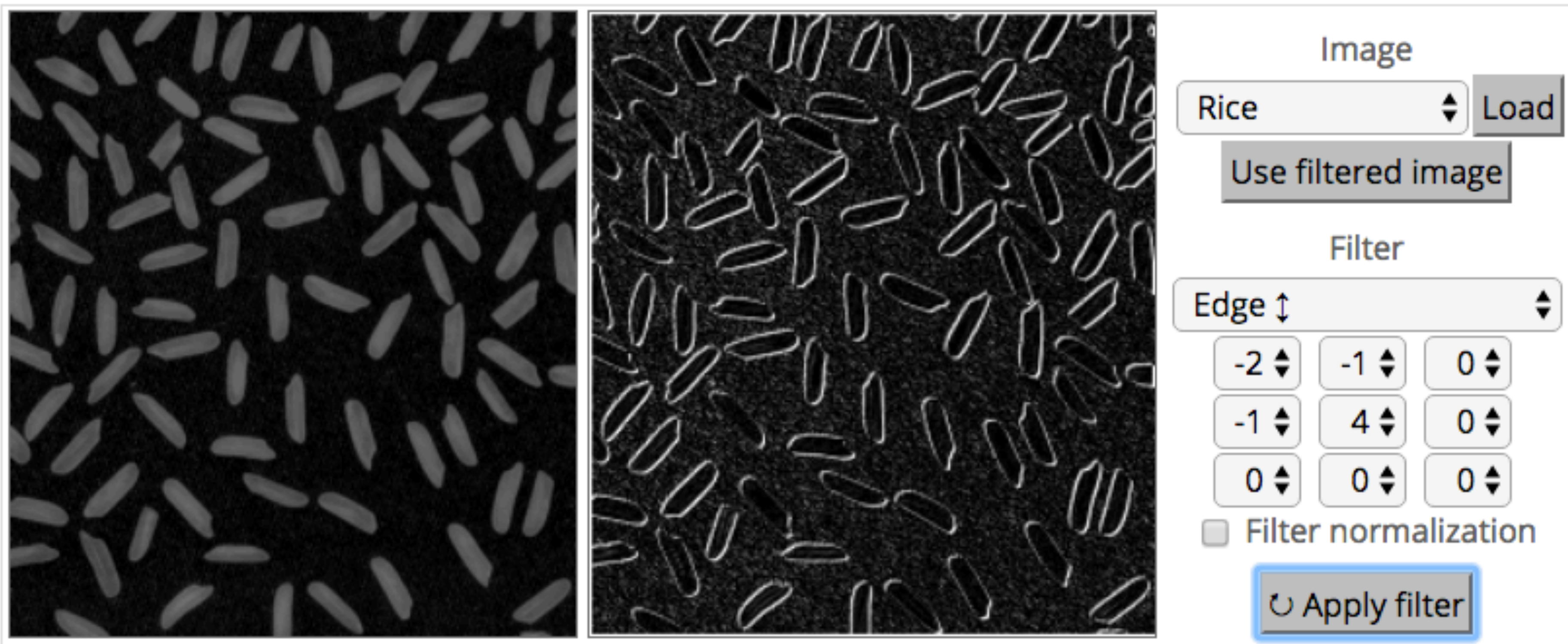
What's a convolution?

http://matlabtricks.com/post-5/3x3-convolution-kernels-with-online-demo



What's a convolution?

http://matlabtricks.com/post-5/3x3-convolution-kernels-with-online-demo

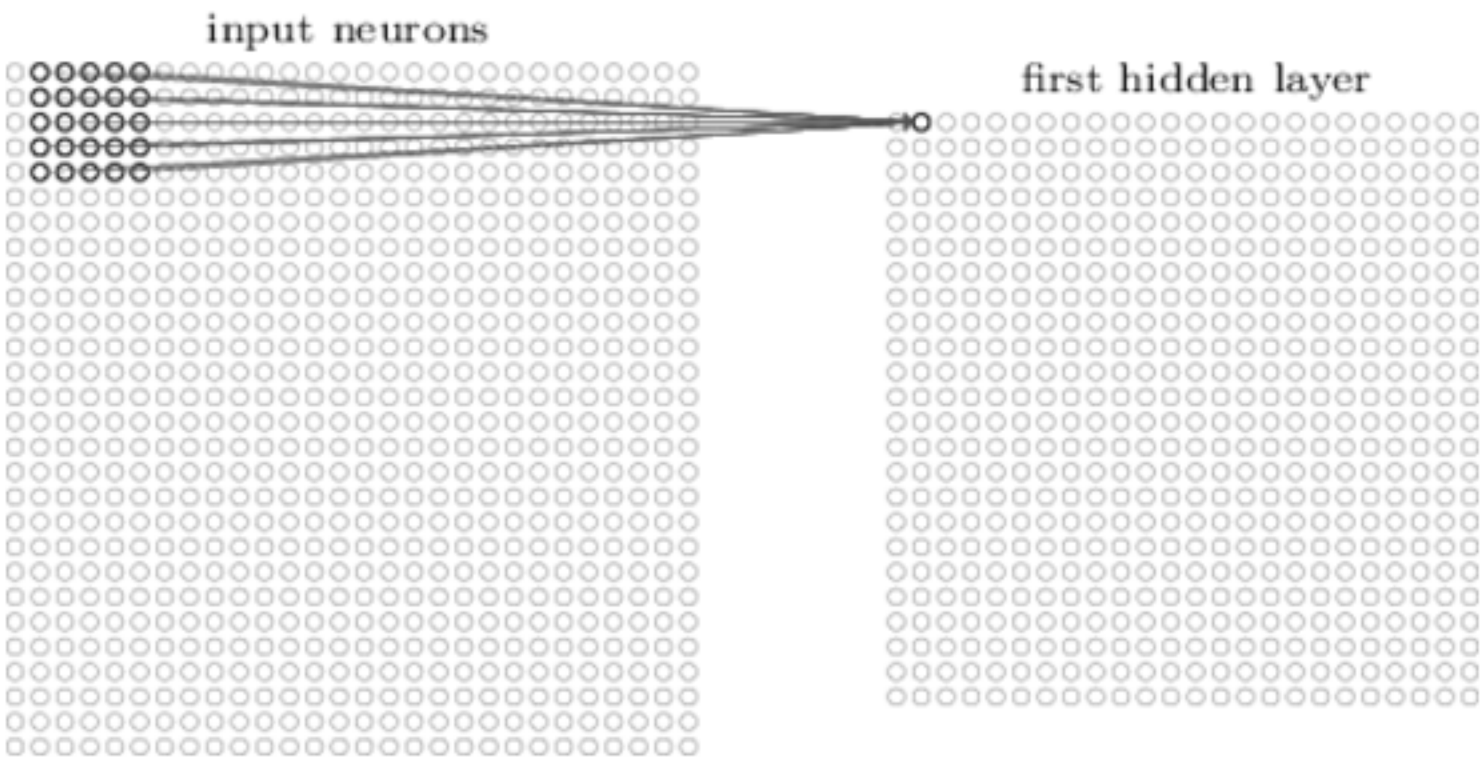


What’s a convolution?

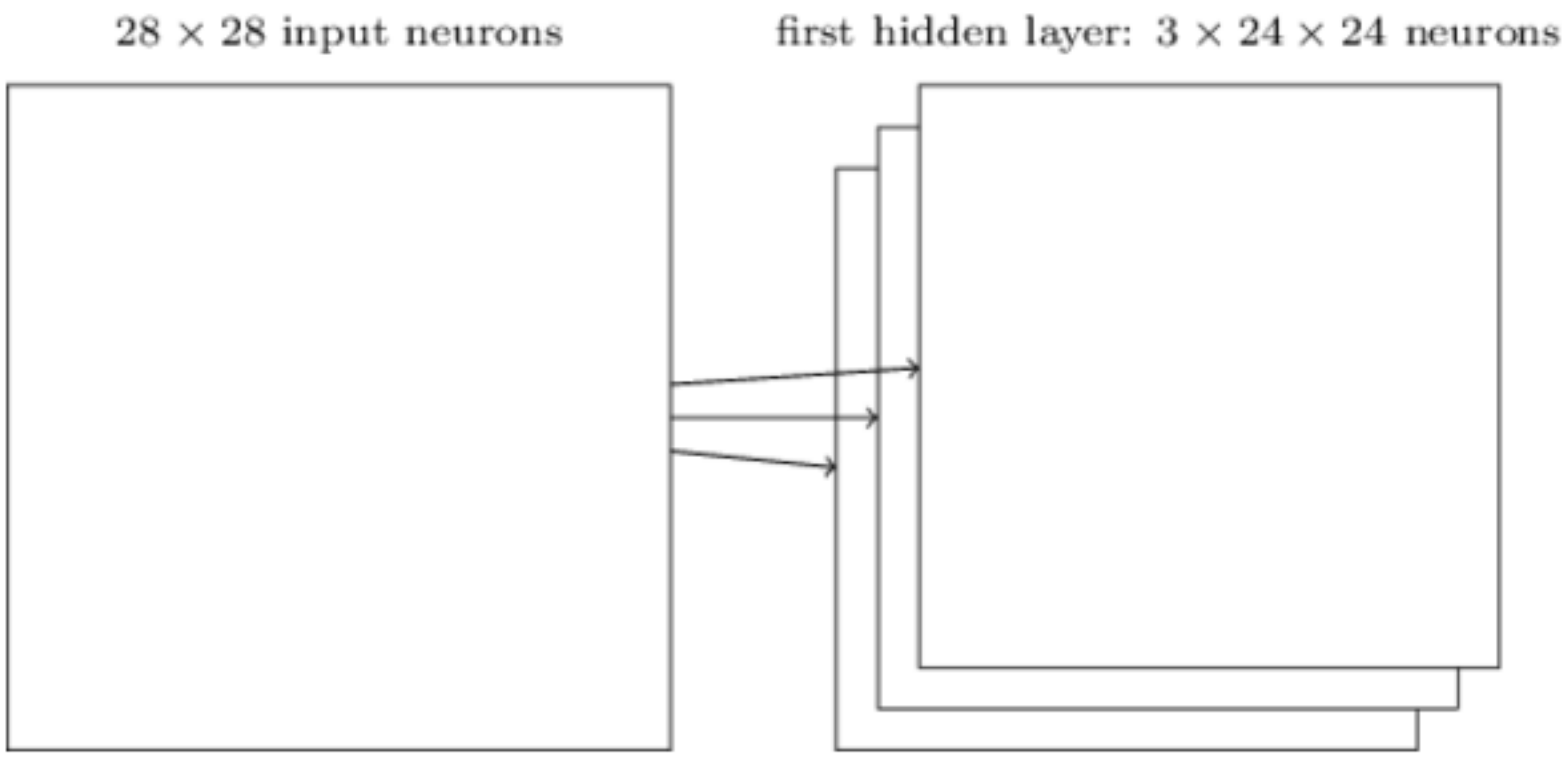
- Basic idea:
  - Pick a 3-3 matrix F of weights
  - Slide this over an image and compute the “inner product” (similarity) of F and the corresponding field of the image, and replace the pixel in the center of the field with the output of the inner product operation
- Key point:
  - Different convolutions extract different types of low-level “features” from an image
  - All that we need to vary to generate these different features is the weights of F

How do we convolve an image with an ANN?

Note that the parameters in the matrix defining the convolution are **tied** across all places that it is used

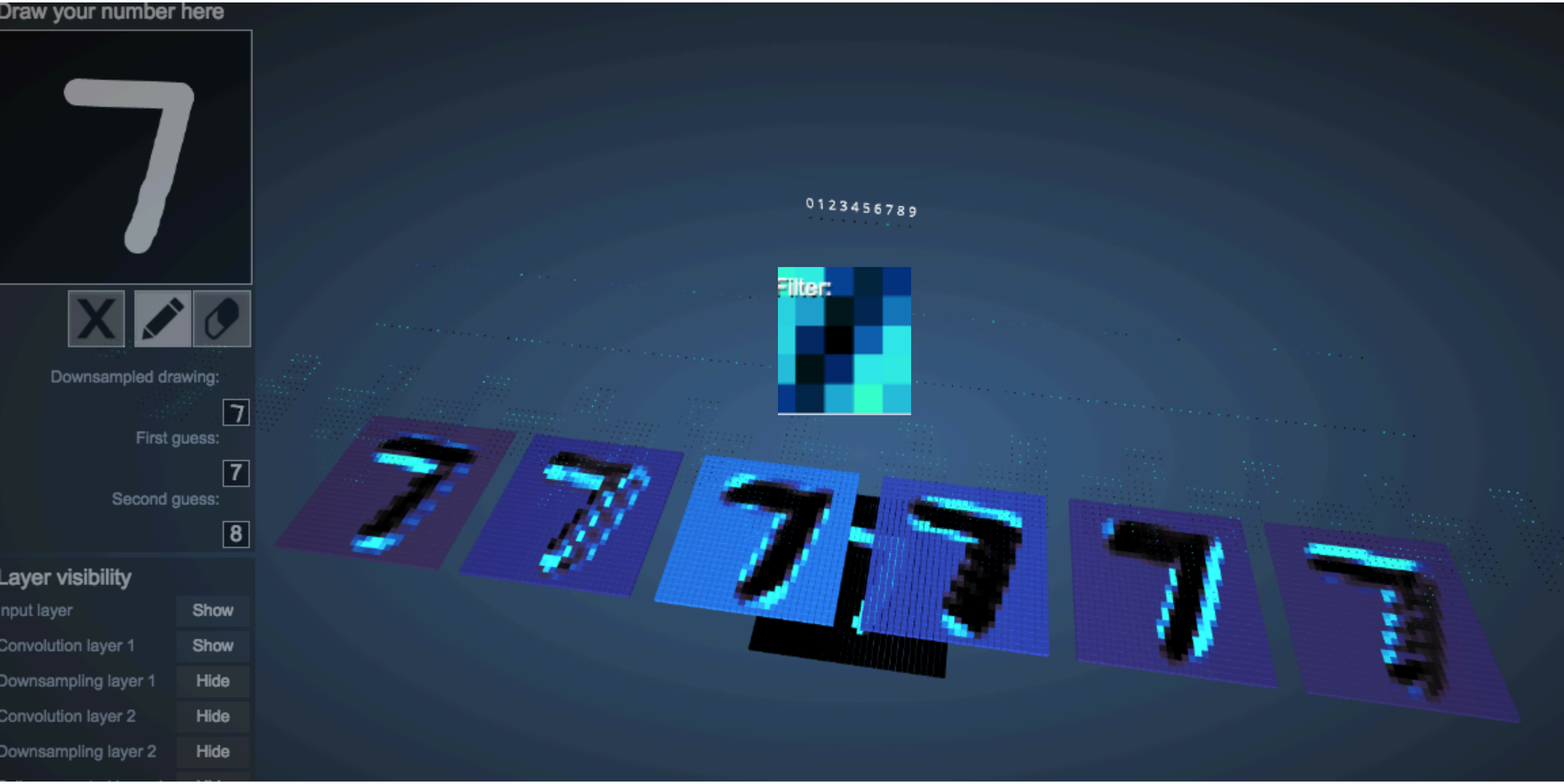


How do we do many convolutions of an image with an ANN?



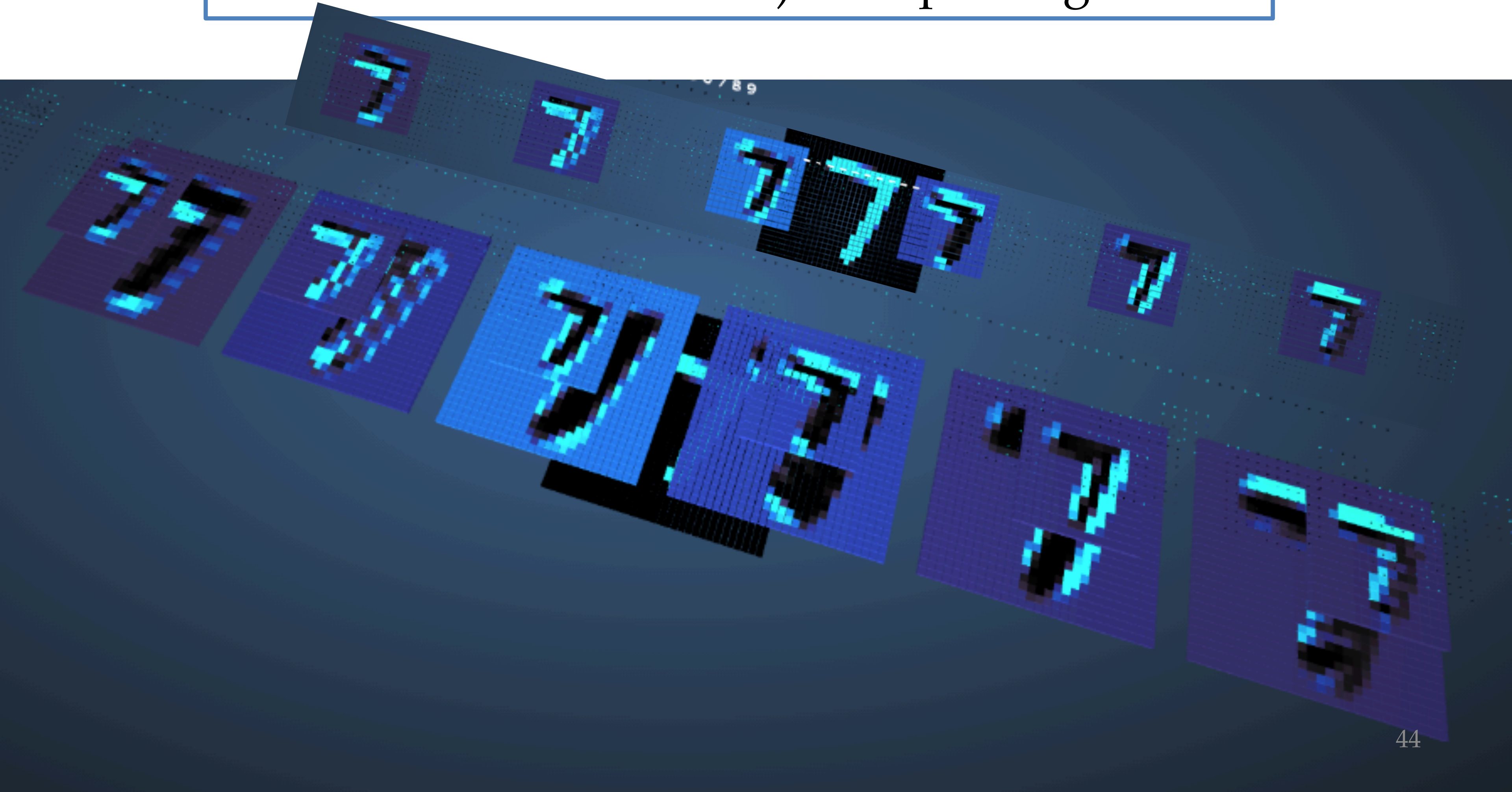
Example: 6 convolutions of a digit

http://scs.ryerson.ca/~aharley/vis/conv/



CNNs typically alternate convolutions, non-linearity, and then downsampling

Downsampling is usually averaging or (more common in recent CNNs) max-pooling



Why do max-pooling?

- Saves space
- Reduces overfitting?
- Because I'm going to add *more* convolutions after it!
  - Allows the short-range convolutions to extend over larger subfields of the images
    - So we can spot larger objects
    - Eg, a long horizontal line, or a corner, or …

PROC. OF THE IEEE, NOVEMBER 1998 7

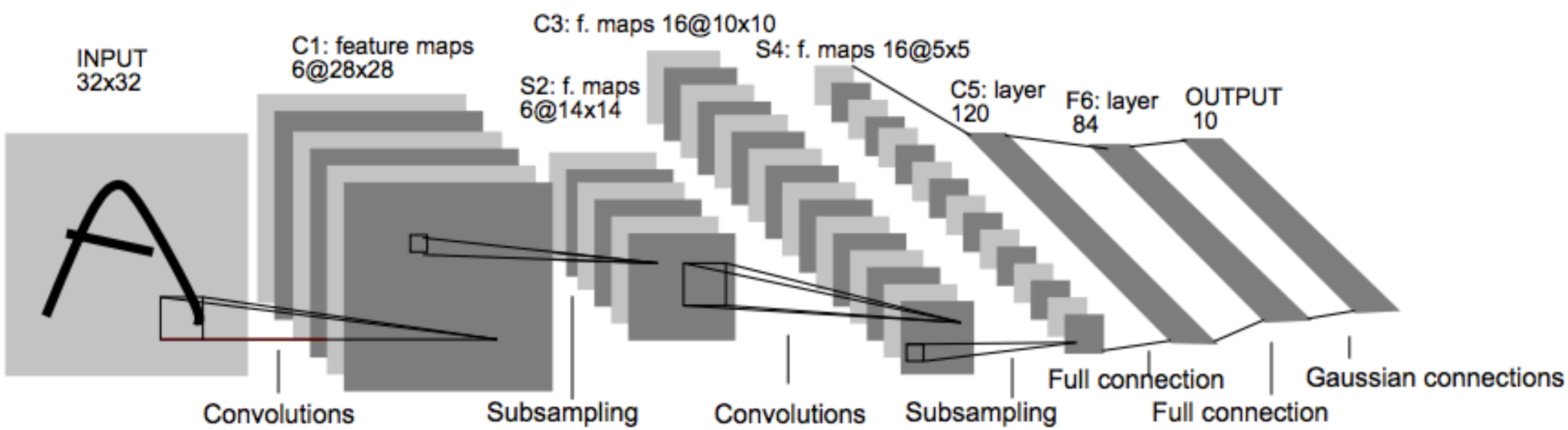
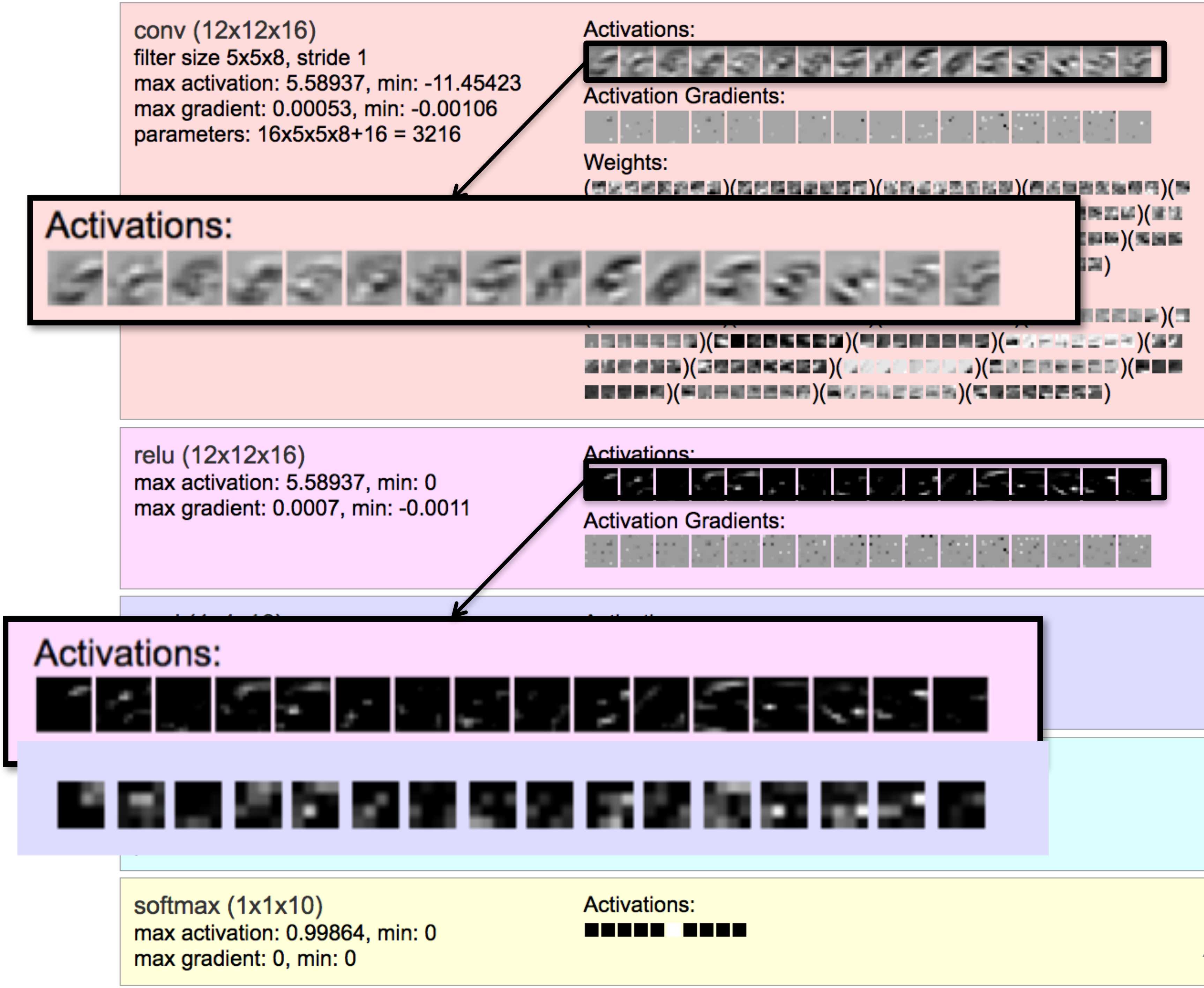


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

Another CNN visualization

https://cs.stanford.edu/people/karpathy/convnetjs/demo/mnist.html


Network Visualization
input (24x24x1)
max activation: 0.99607, min: 0
Activations:
Weights:
Weight Gradients:
filter size 5x5x1, stride 1
max activation: 2.96187, min: -5.43735
max gradient: 0.00068, min: -0.00102
parameters: 8x5x5x1+8 = 208
Activations:
Activation Gradients:
Weights:
Weight Gradients:
Activations:
Activation Gradients:
pool (12x12x8)
pooling size 2x2, stride 2
max activation: 2.96187, min: 0
max gradient: 0.00106, min: -0.00102
Activations:
Activation Gradients:


conv (12x12x16)
filter size 5x5x8, stride 1
max activation: 5.58937, min: -11.45423
max gradient: 0.00053, min: -0.00106
parameters: 16x5x5x8+16 = 3216
Activations:
Activation Gradients:
Weights:
Activations:
relu (12x12x16)
max activation: 5.58937, min: 0
max gradient: 0.0007, min: -0.0011
Activations:
Activation Gradients:
Activations:
softmax (1x1x10)
max activation: 0.99864, min: 0
max gradient: 0, min: 0
Activations:

Why do max-pooling?

- Saves space
- Reduces overfitting?
- Because I’m going to add *more* convolutions after it!
  - Allows the short-range convolutions to extend over larger subfields of the images
    - So we can spot larger objects
    - Eg, a long horizontal line, or a corner, or …
- At some point the feature maps start to get very sparse and blobby – they are indicators of some semantic property, not a recognizable transformation of the image
- Then just use them as features in a “normal” ANN

Why do max-pooling?

- Saves space
- Reduces overfitting?
- Because I'm going to add *more* convolutions after it!
  - Allows the short-range convolutions to extend over larger subfields of the images
    - So we can spot larger objects
    - Eg, a long horizontal line, or a corner, or …

PROC. OF THE IEEE, NOVEMBER 1998 7

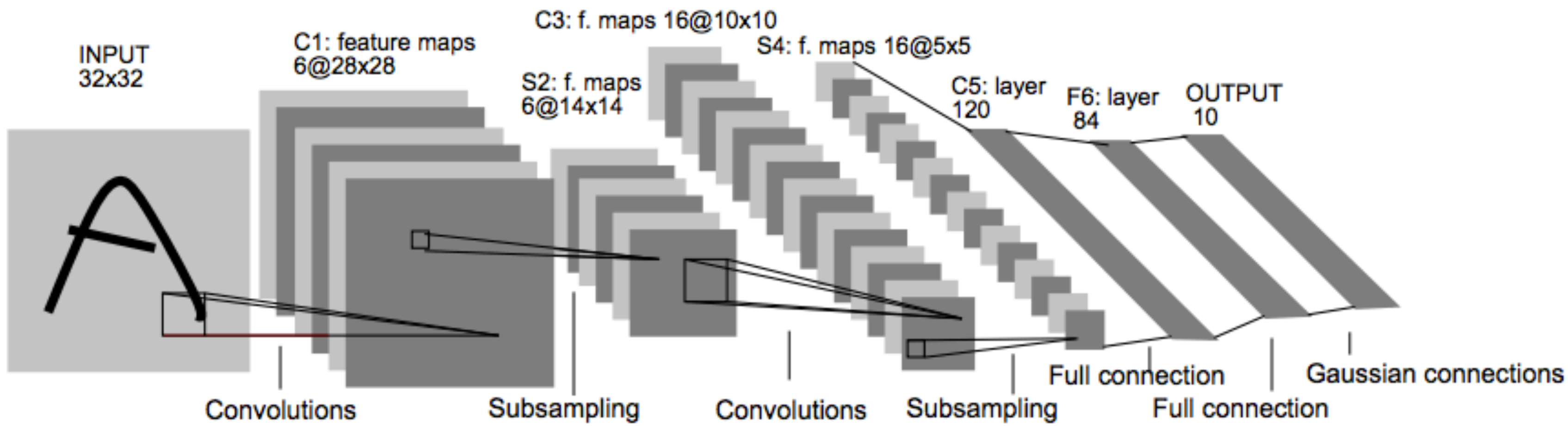
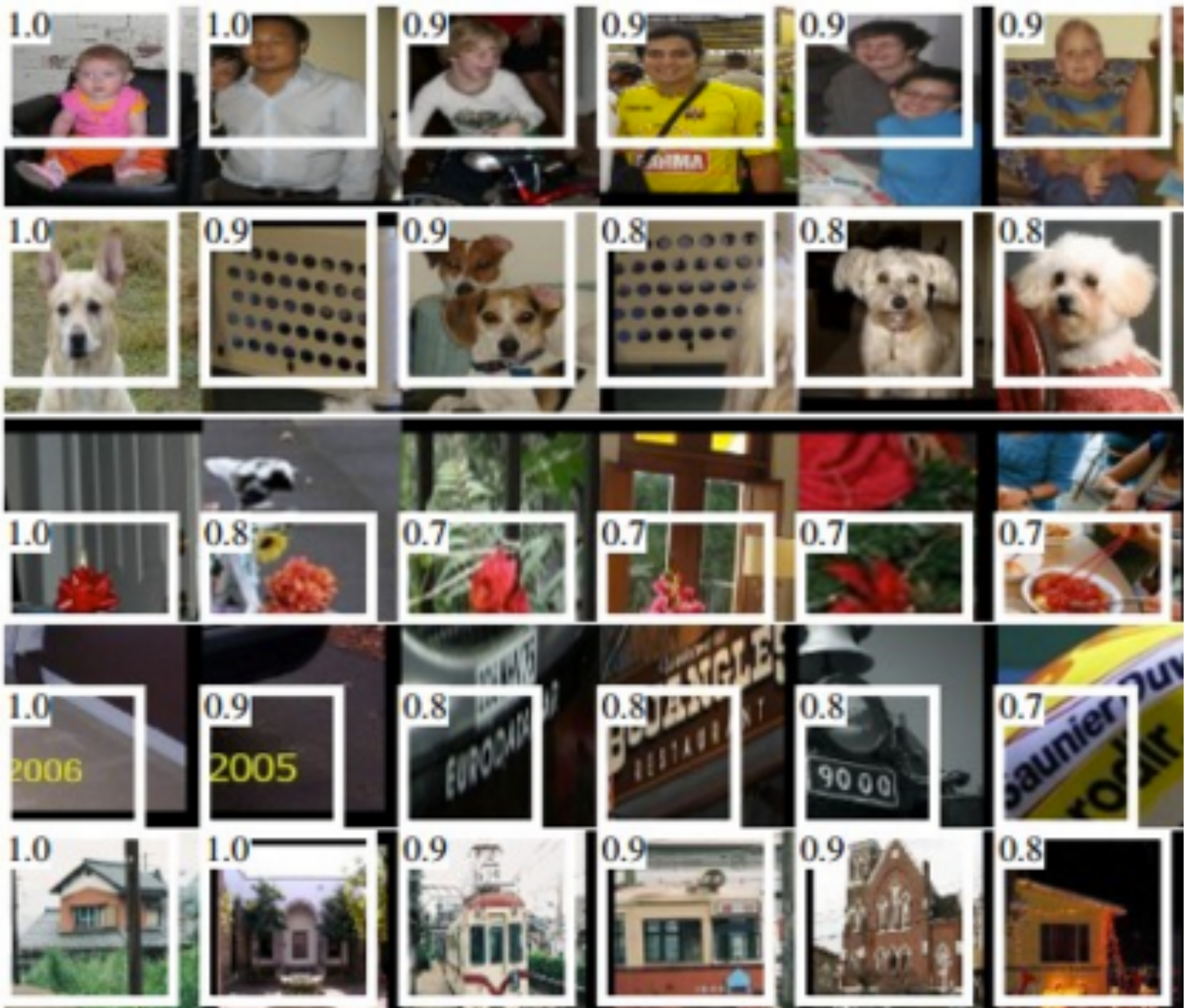


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

Alternating convolution and downsampling



5 layers up

The subfield in a large dataset that gives the strongest output for a neuron

Using RNNs and CNNs

LSTMs can be used for other tasks

encoder/decoder

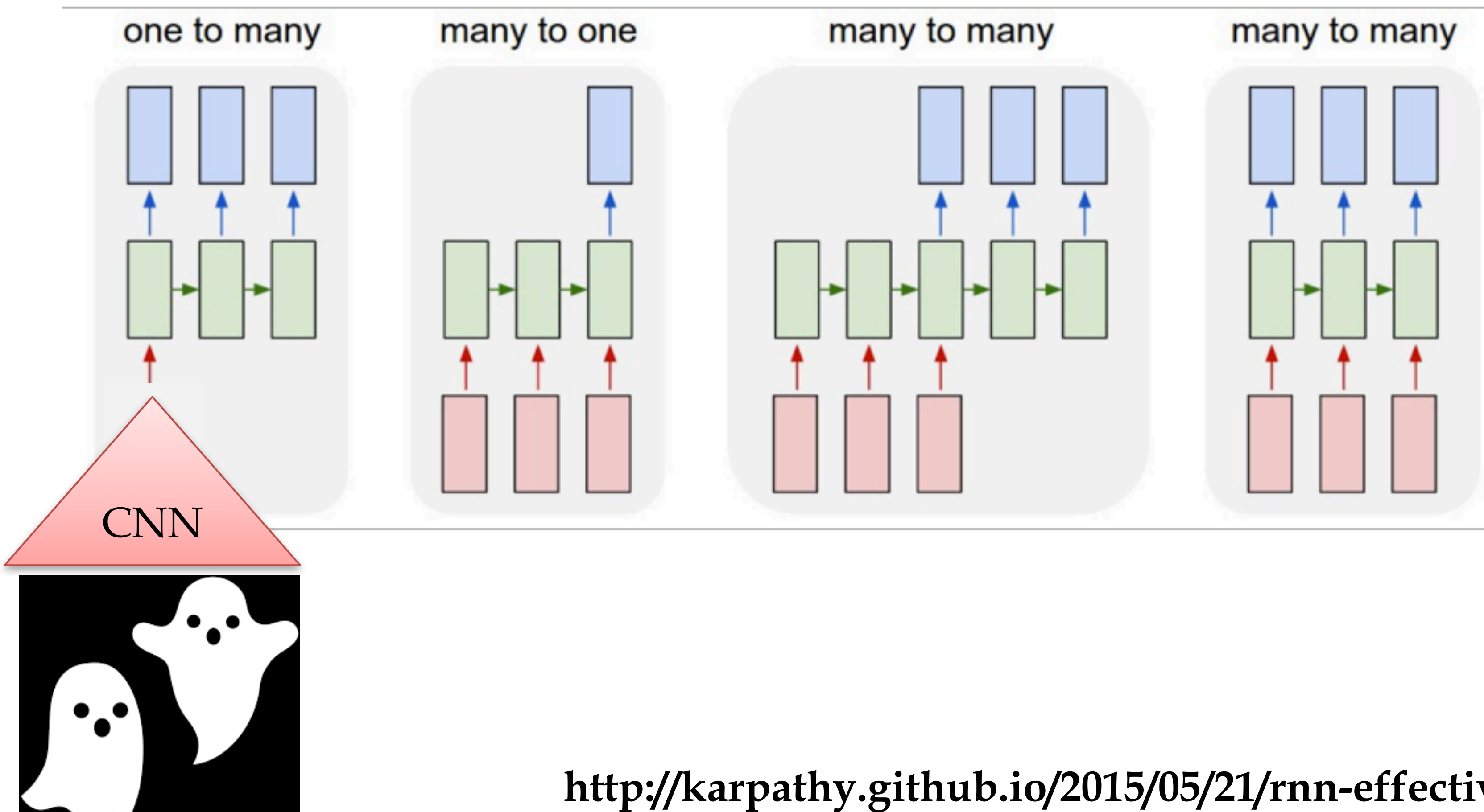
seq2seq

image captioning

sequence classification

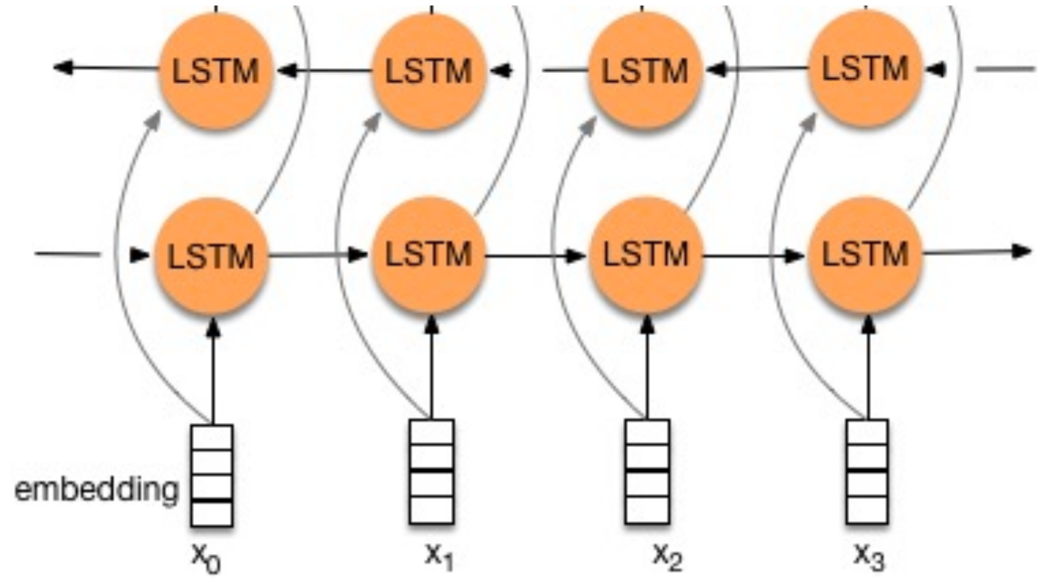
translation

named entity recognition



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

ANN Tricks for NLP



- Common tricks
  - represent words with embeddings
  - represent words in context with RNN hidden state
  - represent a sentence with the last hidden state
    - or pool all hidden states with MAX or SUM
  - biLSTM: run an LSTM in both directions
    - represent with first + last hidden state

  - feed representations into a deeper network….

Example: reasoning about entailment

A large annotated corpus for learning natural language inference

Samuel R. Bowman[*,†]
sbowman@stanford.edu

Gabor Angeli[†,‡]
angeli@stanford.edu

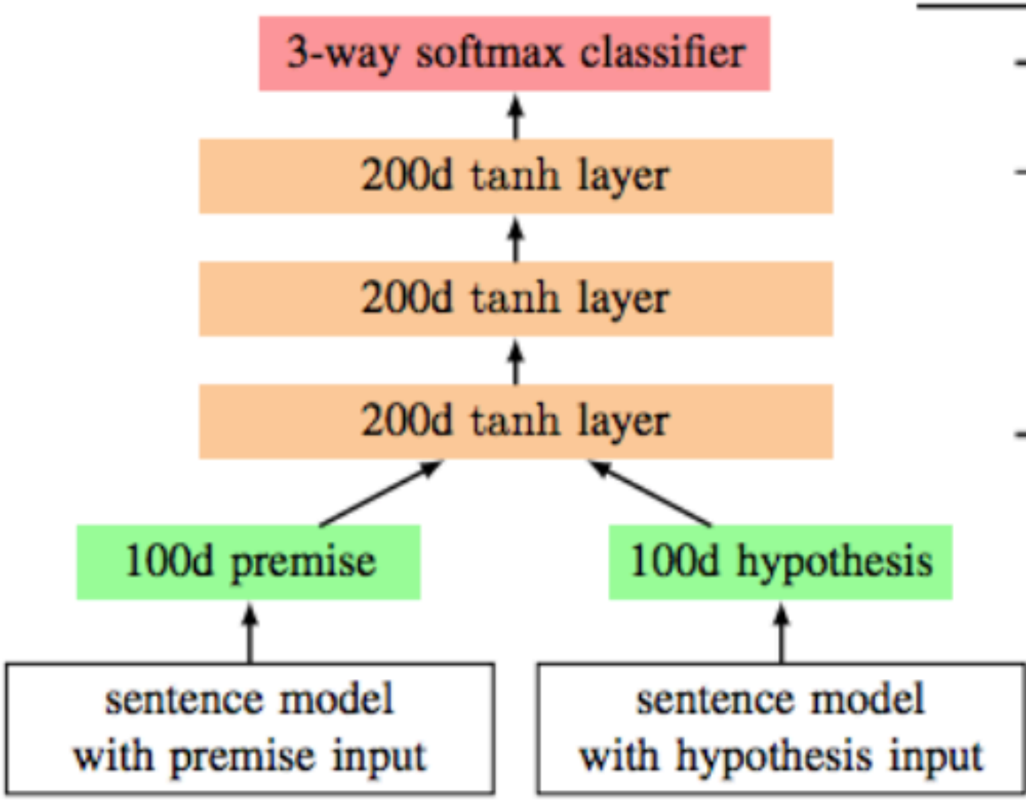
Christopher Potts[*]
cgpotts@stanford.edu

Christopher D. Manning[*,†,‡]
manning@stanford.edu

A man inspects the uniform of a figure in some East Asian country.	**contradiction** C C C C C	The man is sleeping
An older and younger man smiling.	**neutral** N N E N N	Two men are smiling and laughing at the cats playing on the floor.
A black race car starts up in front of a crowd of people.	**contradiction** C C C C C	A man is driving down a lonely road.
A soccer game with multiple males playing.	**entailment** E E E E E	Some men are playing a sport.
A smiling costumed woman is holding an umbrella.	**neutral** N N E C N	A happy woman in a fairy costume holds an umbrella.

RNNs for entailment

Sentence model	Train	Test
100d Sum of words	79.3	75.3
100d RNN	73.1	72.2
100d LSTM RNN	84.8	**77.6**



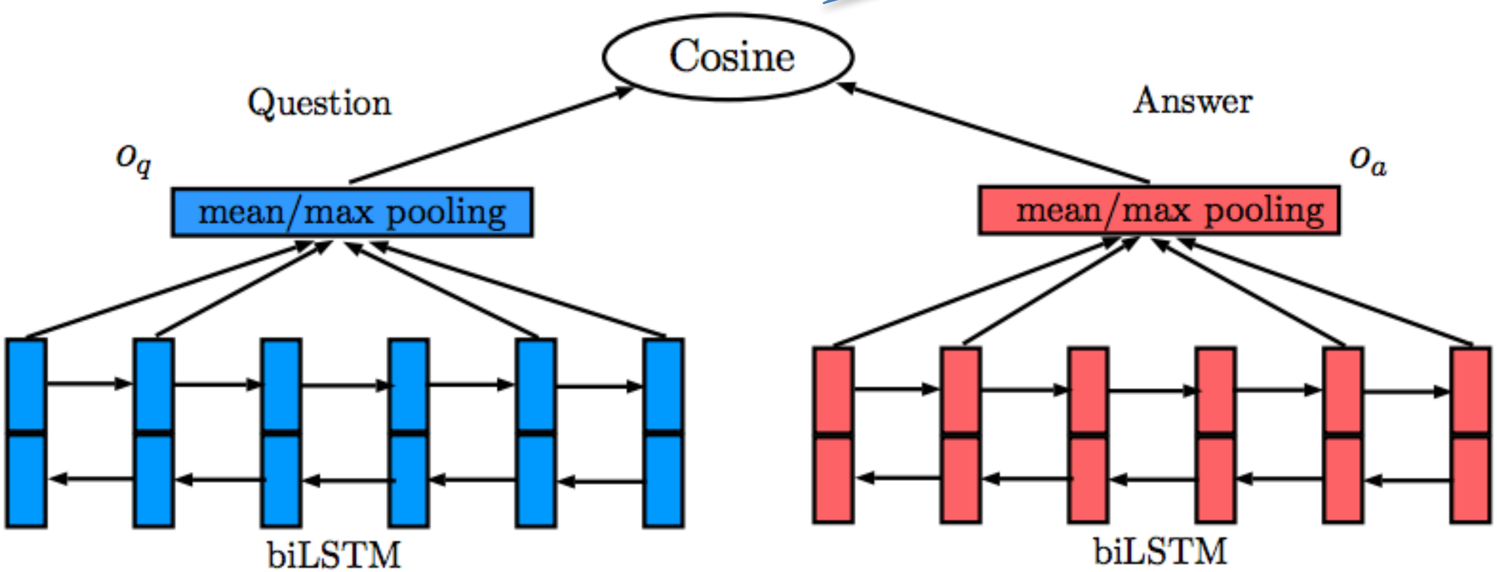
System	SNLI
Edit Distance Based	71.9
Classifier Based	72.2
+ Lexical Resources	**75.0**

Example: question answering

LSTM-BASED DEEP LEARNING MODELS FOR NON-FACTOID ANSWER SELECTION

Ming Tan, Cicero dos Santos, Bing Xiang & Bowen Zhou
IBM Watson Core Technologies
Yorktown Heights, NY, USA
`{mingtan,cicerons,bingxia,zhou}@us.ibm.com`

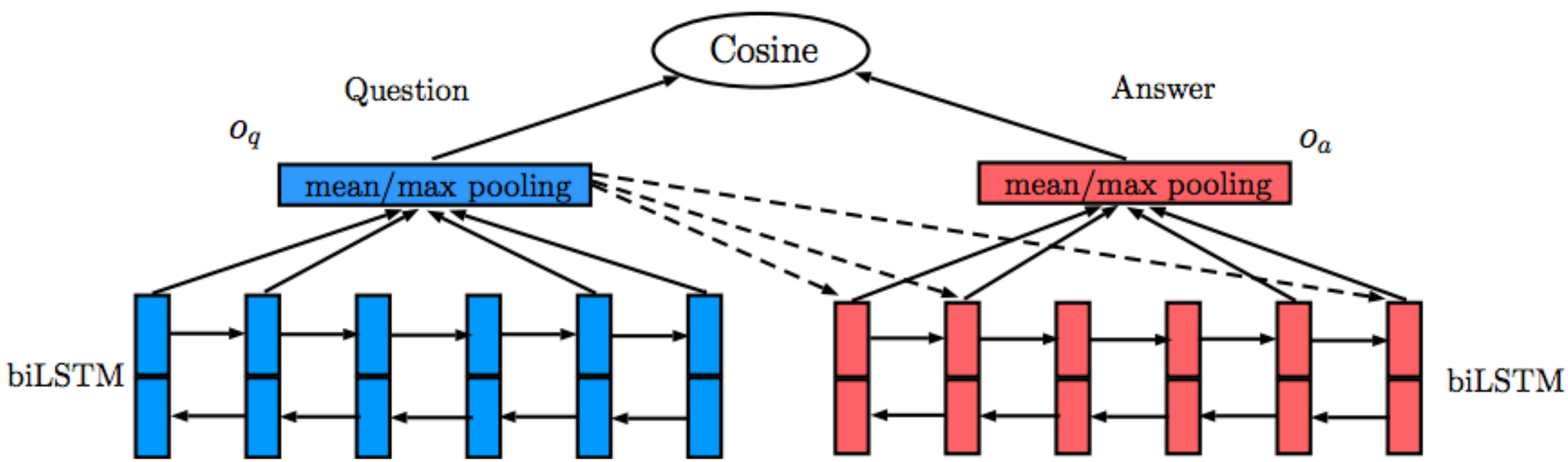
Common trick: train network to make representations similar/dissimilar, not to classify



Example: question answering

Adding **attention**:

- classify the hidden states $\boldsymbol{h}_1, \ldots \boldsymbol{h}_m$ of the answer according to relevance to the question
- when you pool, weight by the classifier's score
- classifier is based on question representation o_q and hidden state $\boldsymbol{h}_i$



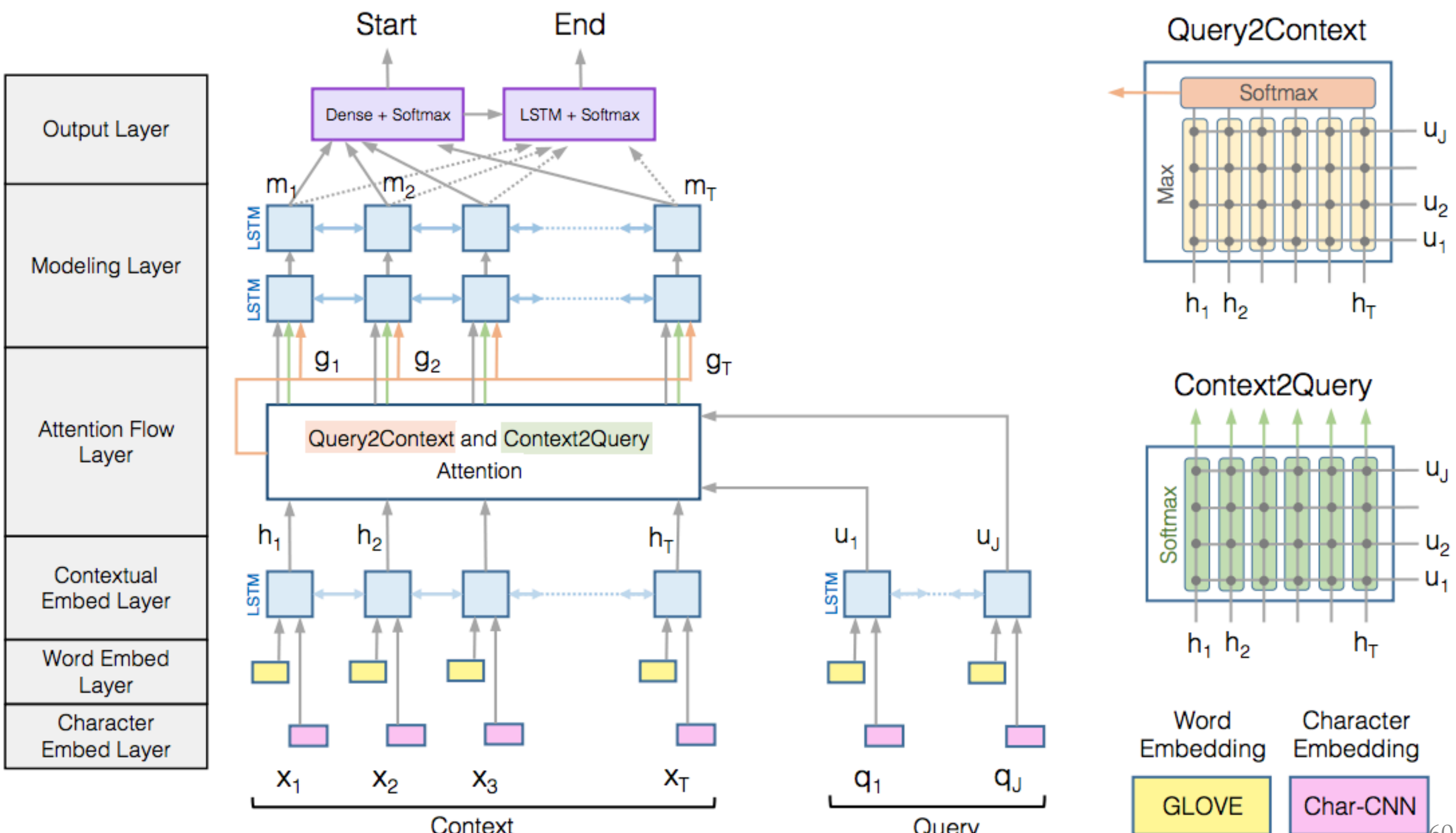
Example: question answering

	Validation	Test1	Test2
A. Bag-of-word	31.9	32.1	32.2
B. Metzler-Bendersky IR model	52.7	55.1	50.8
C. Architecture-II in (Feng et al., 2015)	61.8	62.8	59.2
D. Architecture-II with GESD	**65.4**	**65.3**	**61.0**

	Model	Validation	Test1	Test2
A	QA-LSTM basic-model(head/tail)	54.0	53.1	51.2
B	QA-LSTM basic-model(avg pooling)	58.5	58.2	54.0
C	QA-LSTM basic-model(max pooling)	64.3	63.1	58.0
G	QA-LSTM with attention (max pooling)	66.5	63.7	60.3
H	QA-LSTM with attention (avg pooling)	**68.4**	**68.1**	62.2

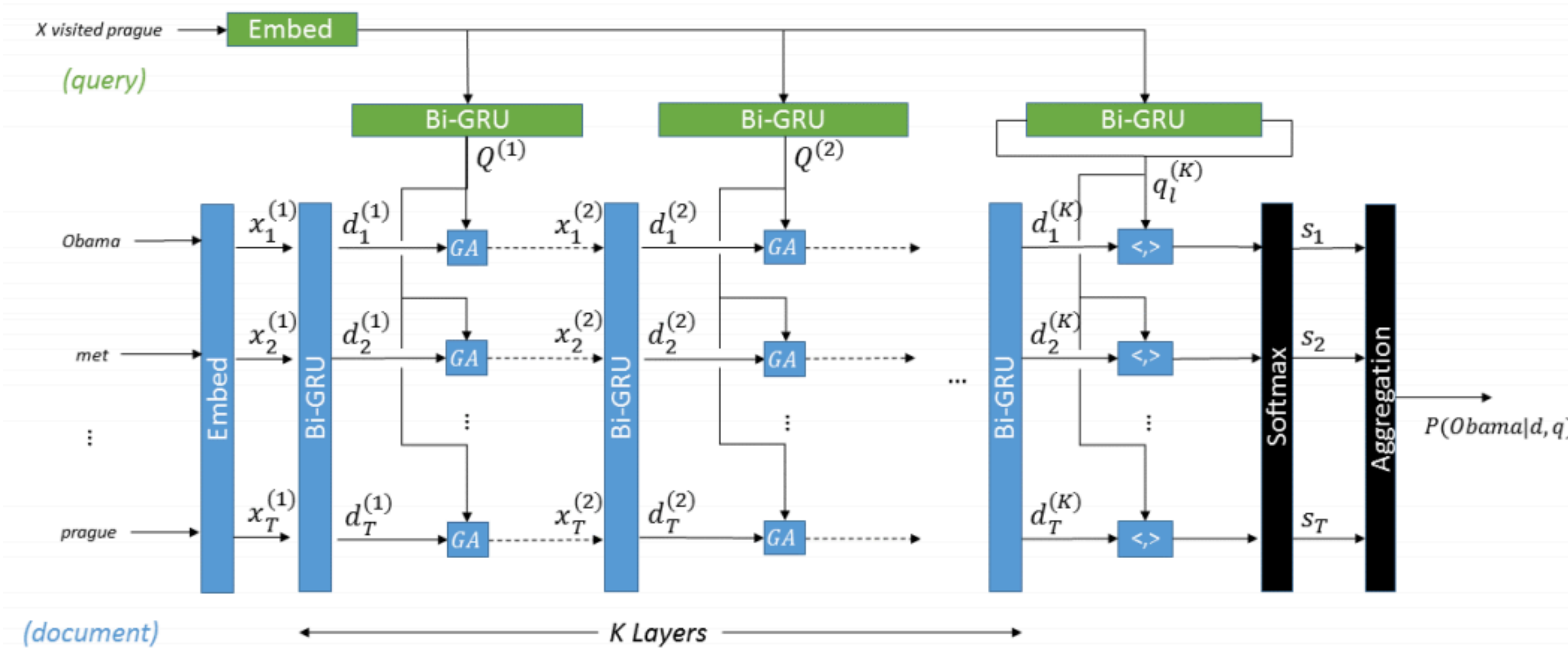
Example: question answering (biDAF)

Seo et al, ICLR 2017



Example: question answering (GA)

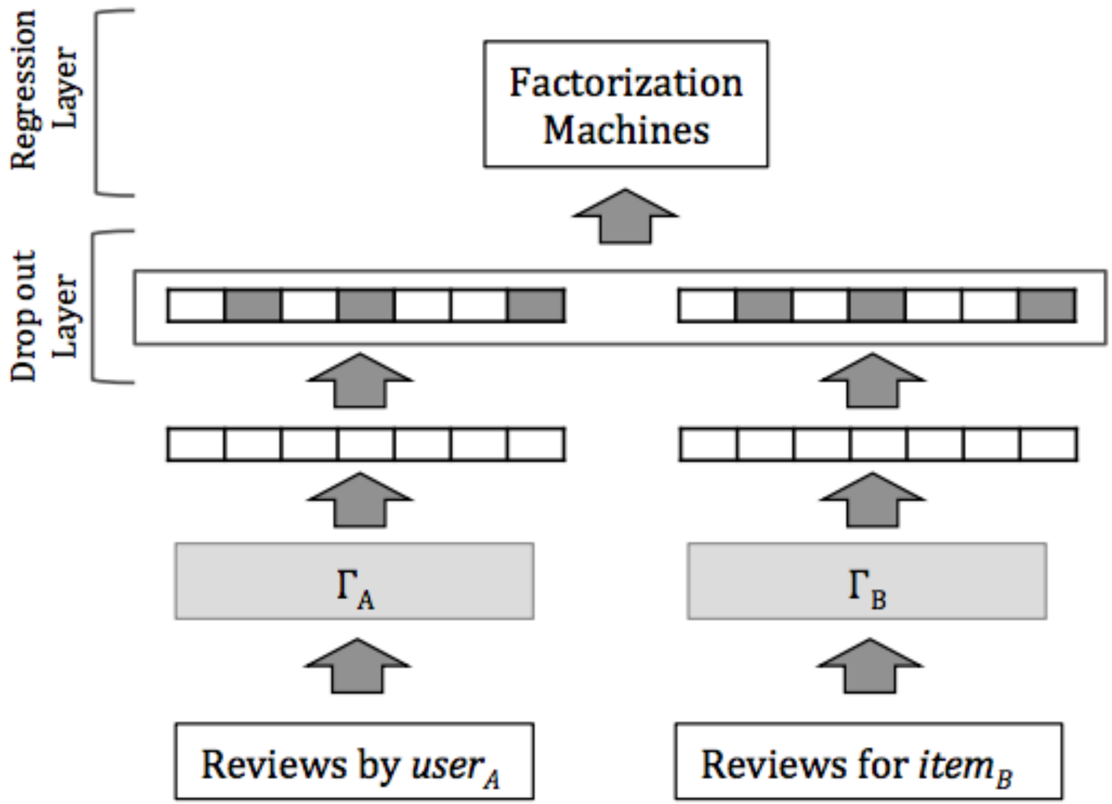
Dhingra, Yang, Cohen, Salakutinof ACL 2017



Model	CNN		Daily Mail		CBT-NE		CBT-CN	
	Val	Test	Val	Test	Val	Test	Val	Test
Humans (query) †	–	–	–	–	–	52.0	–	64.4
Humans (context + query) †	–	–	–	–	–	81.6	–	81.6
LSTMs (context + query) †	–	–	–	–	51.2	41.8	62.6	56.0
Deep LSTM Reader †	55.0	57.0	63.3	62.2	–	–	–	–
Attentive Reader †	61.6	63.0	70.5	69.0	–	–	–	–
Impatient Reader †	61.8	63.8	69.0	68.0	–	–	–	–
MemNets †	63.4	66.8	–	–	70.4	66.6	64.2	63.0
AS Reader †	68.6	69.5	75.0	73.9	73.8	68.6	68.8	63.4
DER Network †	71.3	72.9	–	–	–	–	–	–
Stanford AR (relabeling) †	73.8	73.6	77.6	76.6	–	–	–	–
Iterative Attentive Reader †	72.6	73.3	–	–	75.2	68.6	72.1	69.2
EpiReader †	73.4	74.0	–	–	75.3	69.7	71.5	67.4
AoA Reader †	73.1	74.4	–	–	77.8	72.0	72.2	69.4
ReasoNet †	72.9	74.7	77.6	76.6	–	–	–	–
NSE †	–	–	–	–	78.2	73.2	74.3	**71.9**
BiDAF †	76.3	76.9	80.3	79.6	–	–	–	–
MemNets (ensemble) †	66.2	69.4	–	–	–	–	–	–
AS Reader (ensemble) †	73.9	75.4	78.7	77.7	76.2	71.0	71.1	68.9
Stanford AR (relabeling,ensemble) †	77.2	77.6	80.2	79.2	–	–	–	–
Iterative Attentive Reader (ensemble) †	75.2	76.1	–	–	76.9	72.0	74.1	71.0
EpiReader (ensemble) †	–	–	–	–	76.6	71.8	73.6	70.6
AS Reader (+BookTest) † ‡	–	–	–	–	80.5	76.2	83.2	80.8
AS Reader (+BookTest,ensemble) † ‡	–	–	–	–	*82.3*	*78.4*	*85.7*	*83.7*
GA--	73.0	73.8	76.7	75.7	74.9	69.0	69.0	63.9
GA (update $L(w)$)	**77.9**	**77.9**	**81.5**	**80.9**	76.7	70.1	69.8	67.3
GA (fix $L(w)$)	77.9	77.8	80.4	79.6	77.2	71.4	71.6	68.0
GA (+feature, update $L(w)$)	77.3	76.9	80.7	80.0	77.2	73.3	73.0	69.8
GA (+feature, fix $L(w)$)	76.7	77.4	80.0	79.3	**78.5**	**74.9**	**74.4**	70.7

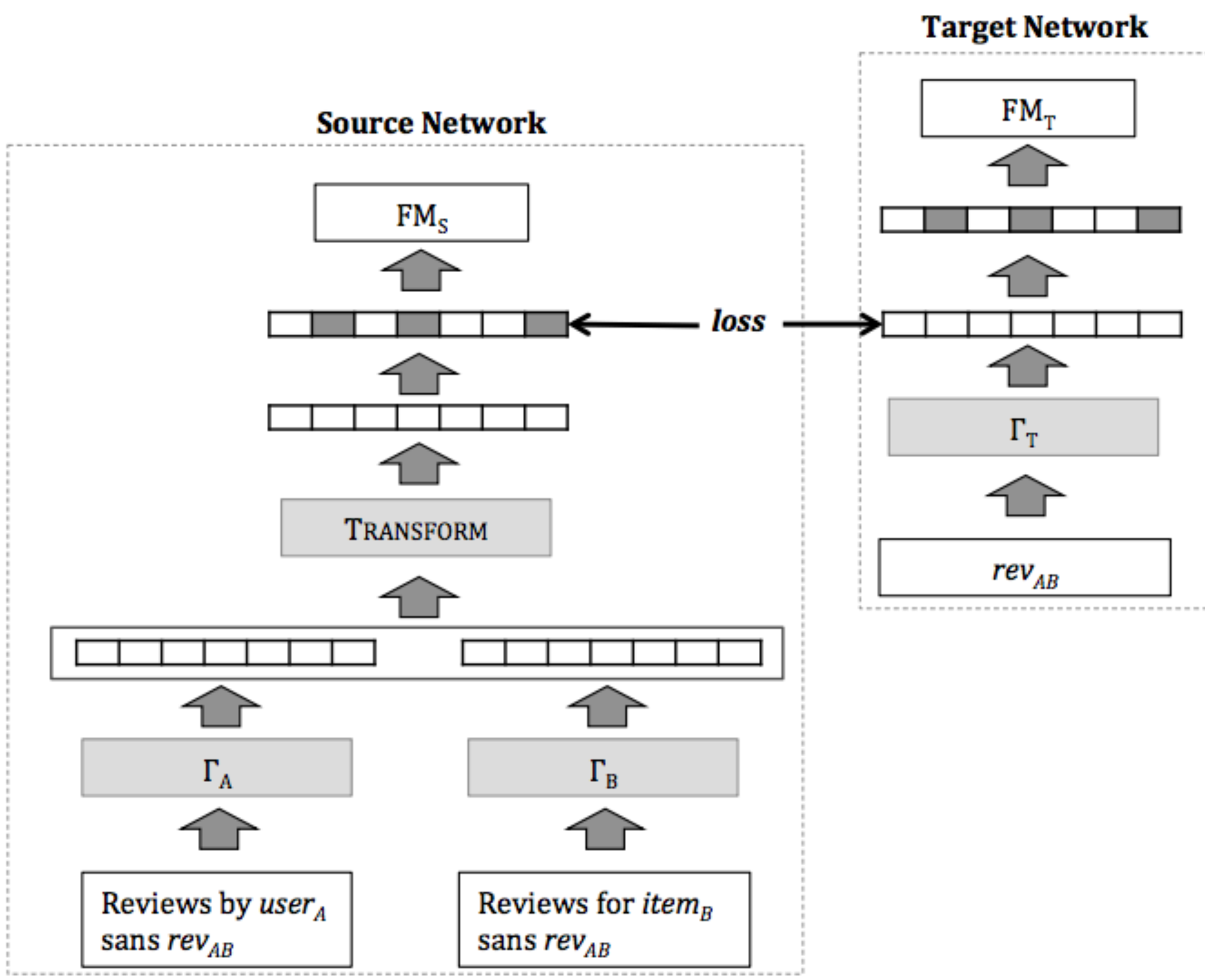
Example: recommendation

Rose Catherine & Cohen, RecSys 2017



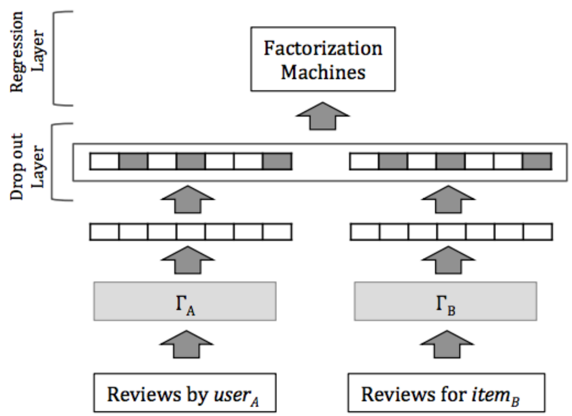
Example: recommendation

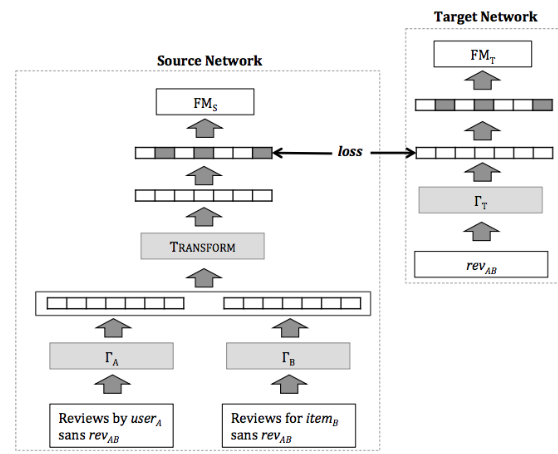
Rose Catherine & Cohen, RecSys 2017



Example: recommendation

Rose Catherine & Cohen, RecSys 2017





Dataset	DeepCoNN + Test Reviews	MF	DeepCoNN	DeepCoNN-rev$_{AB}$	TransNet	TransNet-Ext
Yelp17	1.2106	1.8661	1.8984	1.7045	1.6387	1.5913
AZ-Elec	0.9791	1.8898	1.9704	2.0774	1.8380	1.7781
AZ-CSJ	0.7747	1.5212	1.5487	1.7044	1.4487	1.4780
AZ-Mov	0.9392	1.4324	1.3611	1.5276	1.3599	1.2691