Recurrent Neural Networks

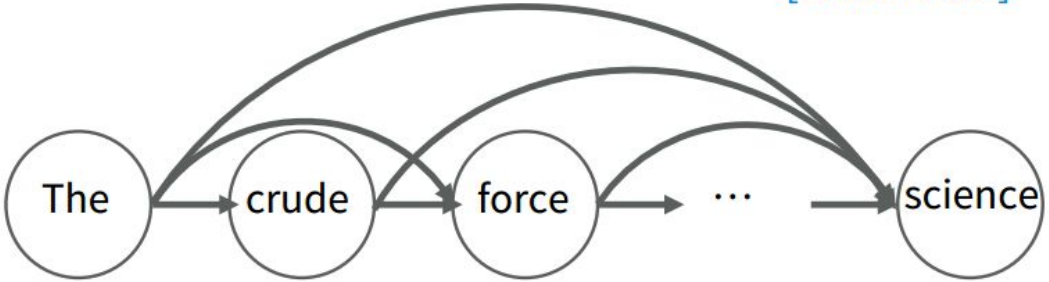
RNN, LSTM, Seq2Seq, NMT and Attention

Aaditya Prakash
March 8, 2017

Language Model

$$p(x_1, x_2, \dots, x_T) = \prod_{t=1}^{T} p(x_t | x_1, \dots, x_{t-1})$$

[Chain rule]



There are way too many histories once you're into a sentence a few words! Exponentially many.

Source: Cho

Language Model - Fix: Markov Assumption

$$p(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} p(x_t | x_1, \ldots, x_{t-1})$$

[Chain rule]

$$\approx \prod_{t=1}^{T} p(x_t | x_{t-n}, \ldots, x_{t-1})$$

Problem: Very small window gives bad prediction
Solution: Smoothing, attention (discussed later)

Source: Cho

Language Model - Recurrent Model

$$p(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} p(x_t | x_1, \ldots, x_{t-1})$$

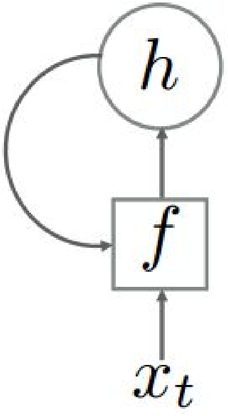
[Chain rule]

Recursive construction of f

1. Initialization $h_0 = 0$
2. Recursion $h_t = f(x_t, h_{t-1})$

We call h_t a hidden state or memory

h_t summarizes the history $(x_1, \ldots, x_t)$



Source: Cho

Language Model - Recurrent Model

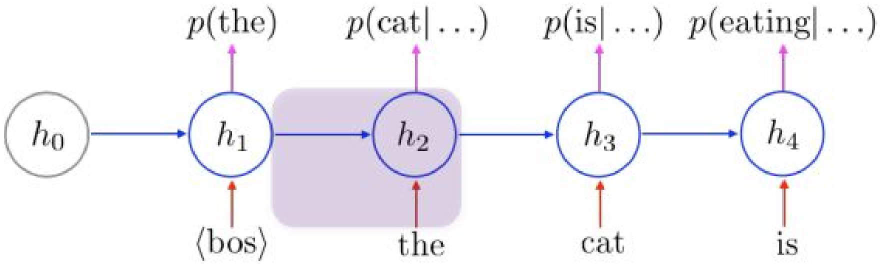
Transition Function $h_t = f(h_{t-1}, x_t)$

Inputs

i. Current word $x_t \in \{1, 2, \ldots, |V|\}$
ii. Previous state $h_{t-1} \in \mathbb{R}^d$

Parameters

i. Input weight matrix $W \in \mathbb{R}^{|V| \times d}$
ii. Transition weight matrix $U \in \mathbb{R}^{d \times d}$
iii. Bias vector $b \in \mathbb{R}^d$



Source: Cho

Language Model - Recurrent Model

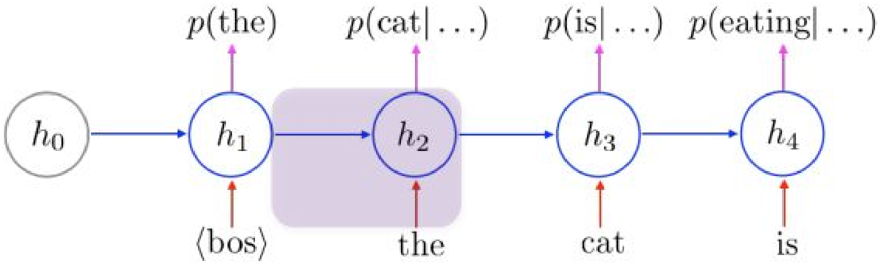
Naïve Transition Function

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

Element-wise nonlinear transformation

Trainable word vector

Linear transformation of previous state



Source: Cho

Language Model - Recurrent Model

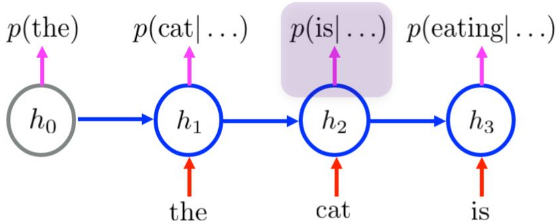
Prediction Function $p(x_{t+1} = w|x_{\leq t}) = g_w(h_t)$

$$p(x_{t+1} = w|x_{\leq t}) = g_w(h_t) = \frac{\exp(R[w]^\top h_t + c_w)}{\sum_{i=1}^{|V|} \exp(R[i]^\top h_t + c_i)}$$

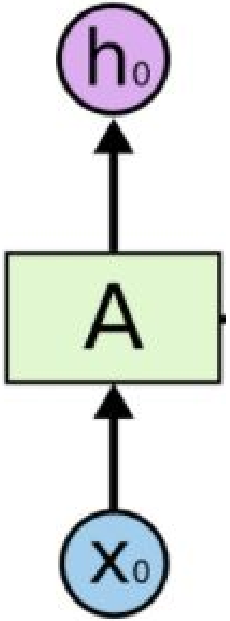
Compatibility between trainable word vector and hidden state

Normalize

Exponentiate

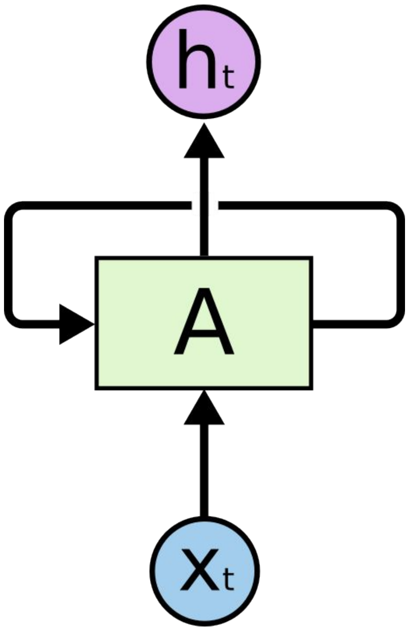


Artificial Neuron



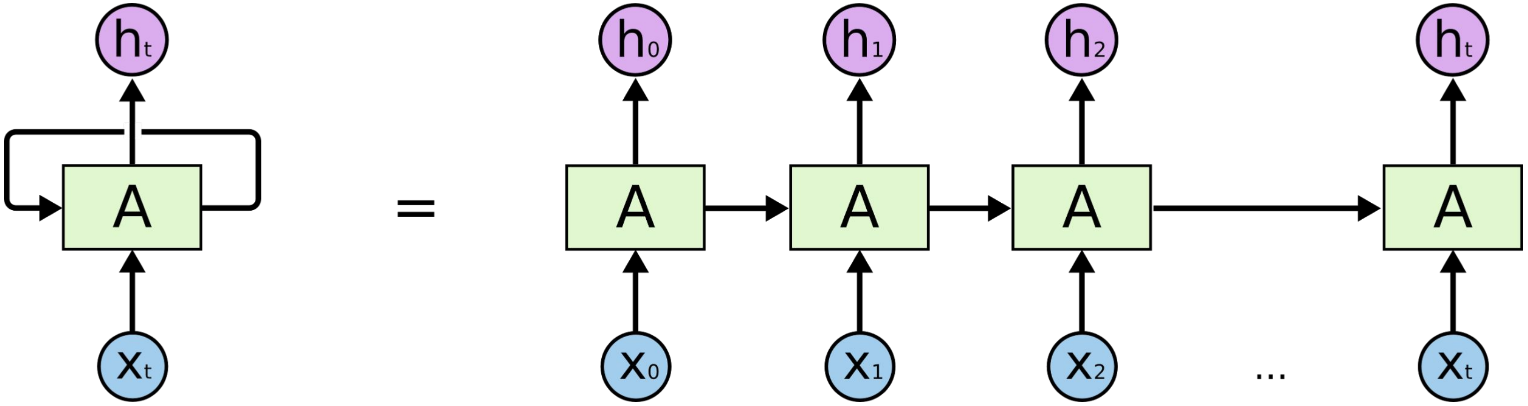
Source: Colah

Recurrent Neuron



Source: Colah

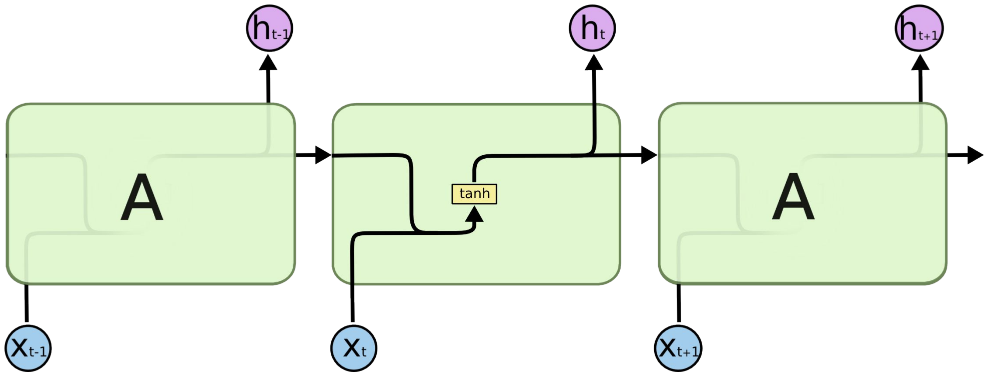
Recurrent Neuron - Unrolled

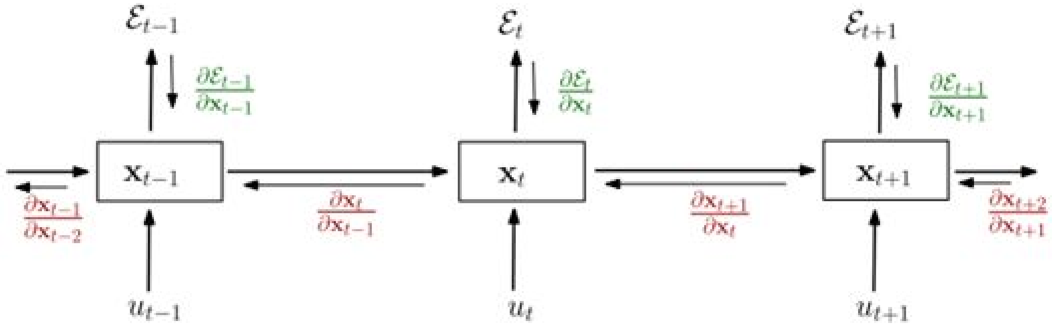


An unrolled recurrent neural network.

Source: Colah

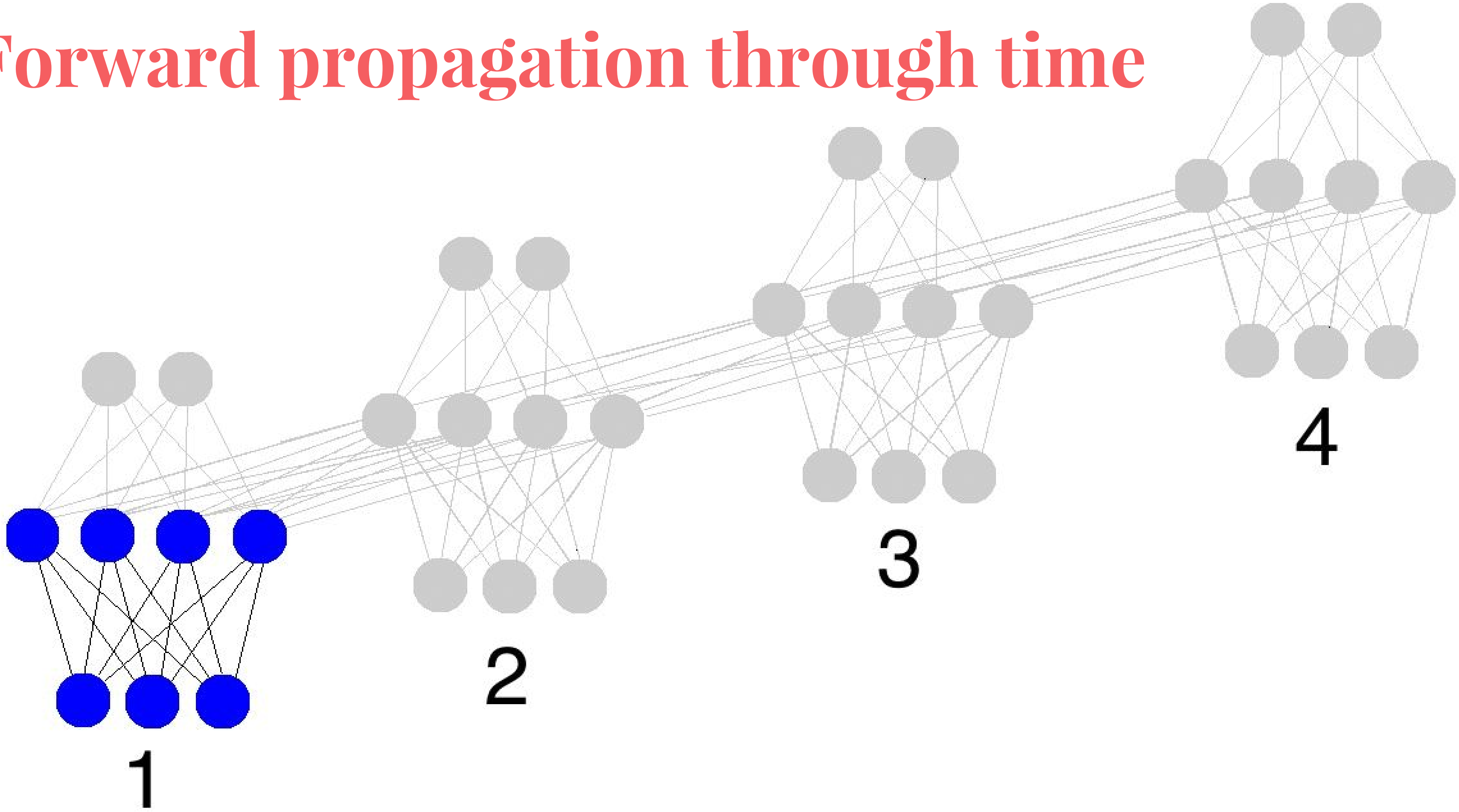
RNN - Structure





Source: Colah

Forward propagation through time



Backpropagation through time

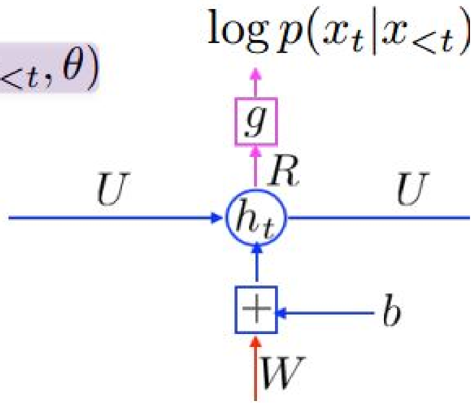
How do we compute $\nabla\mathcal{L}(\theta, D)$ *?*

- Cost as a sum of per-sample cost function

$$\nabla\mathcal{L}(\theta, D) = \sum_{X \in D} \nabla\mathcal{L}(\theta, X)$$

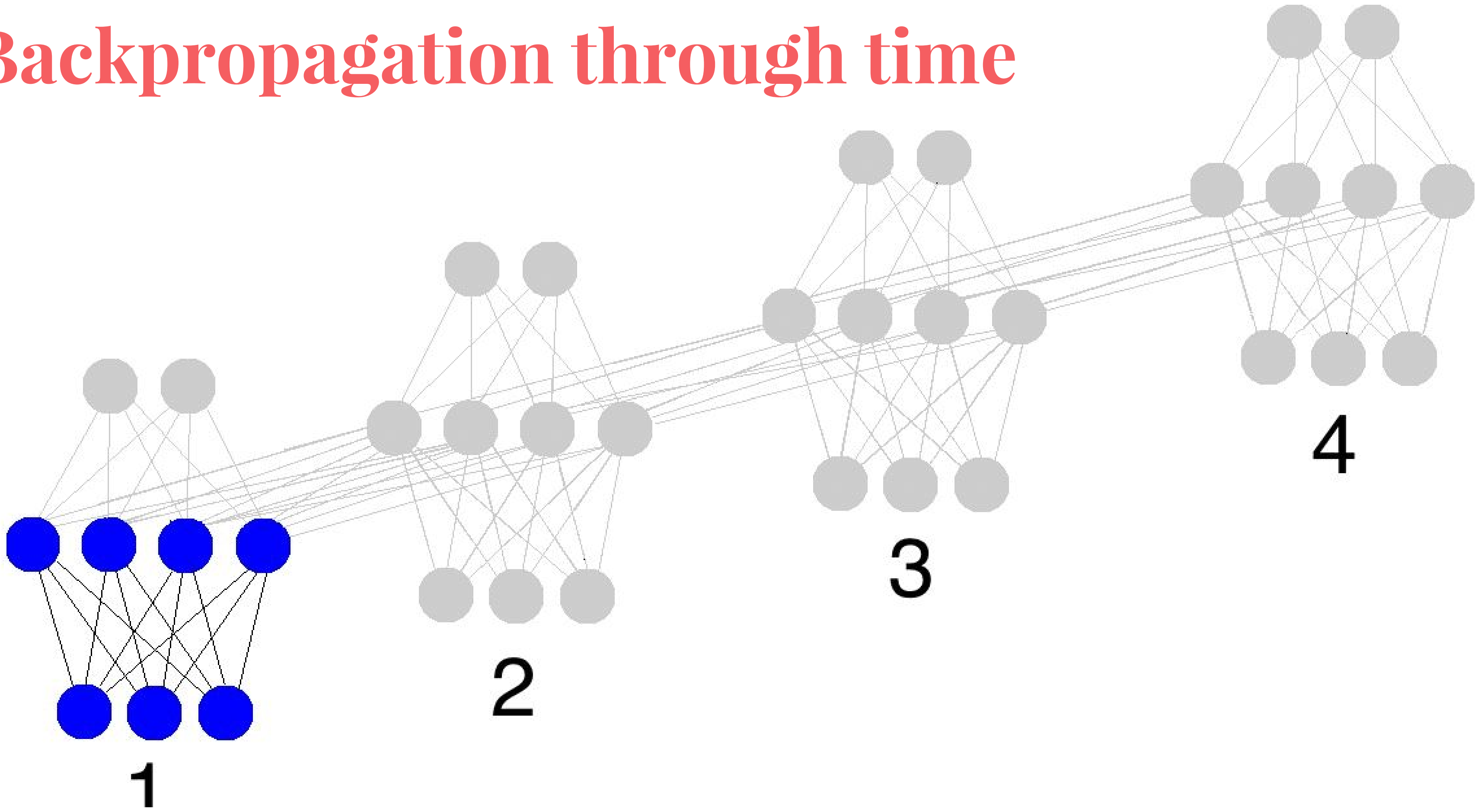
- Per-sample cost as a sum of per-step cost functions

$$\nabla\mathcal{L}(\theta, X) = \sum_{t=1}^{T} \nabla \log p(x_t | x_{<t}, \theta)$$



Source: Cho

Backpropagation through time



BPTT

1. Measure the influence of the past on the future

$$\frac{\partial \log p(x_{t+n}|x_{<t+n})}{\partial h_t} = \frac{\partial \log p(x_{t+n}|x_{<t+n})}{\partial g} \frac{\partial g}{\partial h_{t+n}} \frac{\partial h_{t+n}}{\partial h_{t+n-1}} \cdots \frac{\partial h_{t+1}}{\partial h_t}$$

2. With a naïve transition function

$$f(h_{t-1}, x_{t-1}) = \tanh(W\left[x_{t-1}\right] + U h_{t-1} + b)$$

We get $\dfrac{\partial J_{t+n}}{\partial h_t} = \dfrac{\partial J_{t+n}}{\partial g} \dfrac{\partial g}{\partial h_{t+N}} \prod_{n=1}^{N} U^\top \operatorname{diag}\left(\dfrac{\partial \tanh(a_{t+n})}{\partial a_{t+n}}\right)$

Source: Cho

BPTT

1. Measure the influence of the past on the future

$$\frac{\partial \log p(x_{t+n}|x_{<t+n})}{\partial h_t} = \frac{\partial \log p(x_{t+n}|x_{<t+n})}{\partial g} \frac{\partial g}{\partial h_{t+n}} \frac{\partial h_{t+n}}{\partial h_{t+n-1}} \cdots \frac{\partial h_{t+1}}{\partial h_t}$$

2. With a naïve transition function

$$f(h_{t-1}, x_{t-1}) = \tanh(W[x_{t-1}] + U h_{t-1} + b)$$

We get $$\frac{\partial J_{t+n}}{\partial h_t} = \frac{\partial J_{t+n}}{\partial g} \frac{\partial g}{\partial h_{t+N}} \underbrace{\prod_{n=1}^{N} U^\top \mathrm{diag}\left(\frac{\partial \tanh(a_{t+n})}{\partial a_{t+n}}\right)}_{\textit{Problematic!}}$$

Source: Cho

RNN: vanishing & exploding gradient

- What happens?

$$\frac{\partial J_{t+n}}{\partial h_t} = \frac{\partial J_{t+n}}{\partial g} \frac{\partial g}{\partial h_{t+N}} \underbrace{\prod_{n=1}^{N} U^\top \mathrm{diag}\left(\frac{\partial \tanh(a_{t+n})}{\partial a_{t+n}}\right)}$$

1. The gradient *likely* explodes if

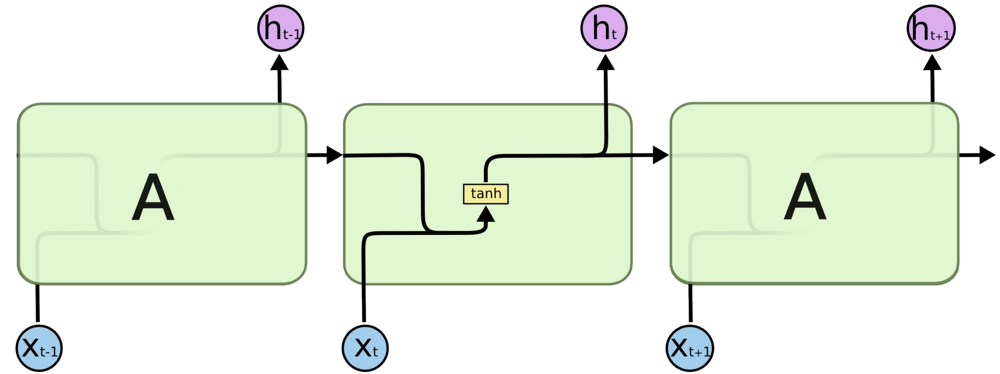
$$e_{\max} \geq \frac{1}{\max \tanh'(x)} = 1$$

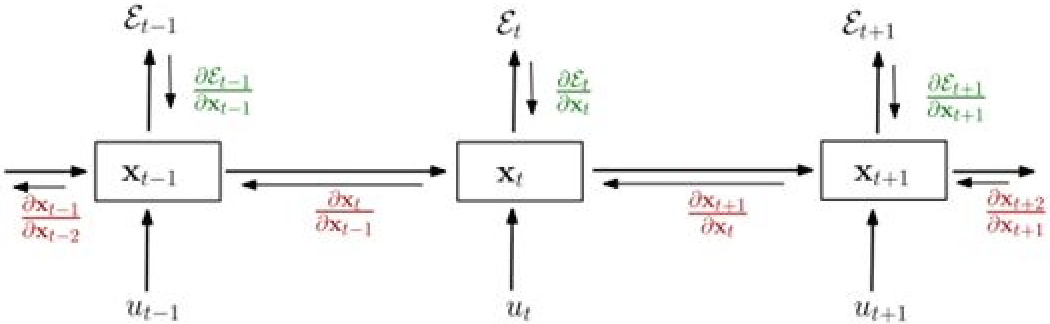
2. The gradient *likely* vanishes if

$$e_{\max} < \frac{1}{\max \tanh'(x)} = 1,$$ where $e_{\max}$: largest eigenvalue of U

Source: Cho

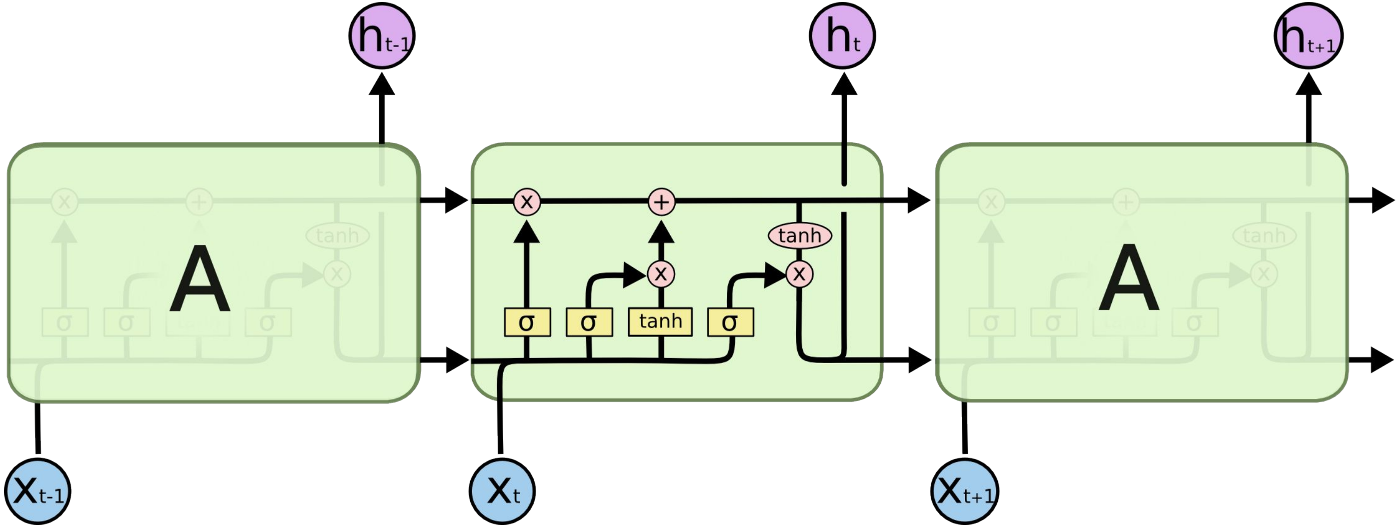
RNN - Structure





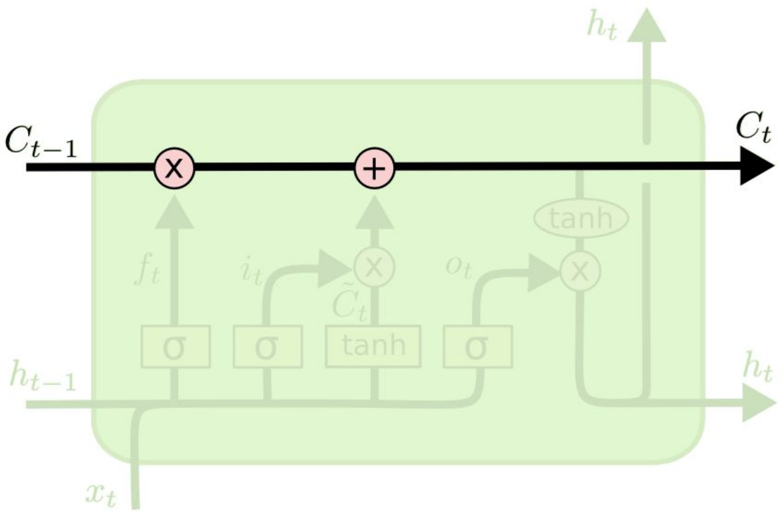
Source: Colah

Solution - Long Memory and Short Memory



Source: Colah

Solution - Long Memory and Short Memory

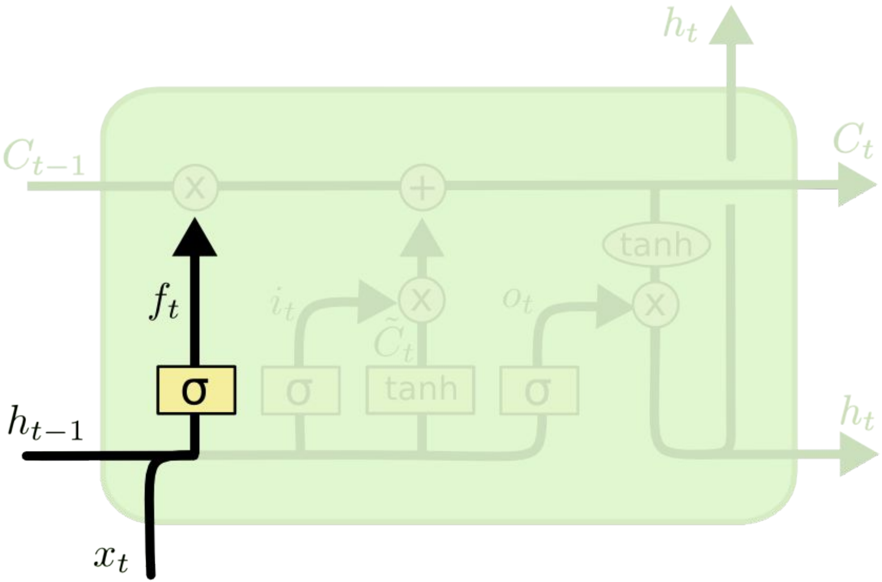


Cell State

Why state? E.g remember gender, so that proper pronoun can be used.

Source: Colah

Solution - Long Memory and Short Memory



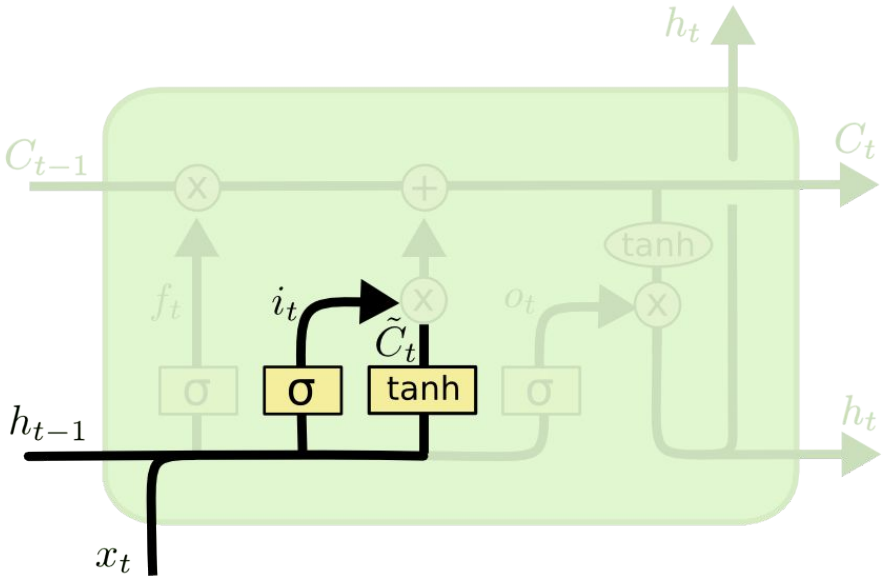
$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

Forget Gate Layer

Why forget then? Perhaps new subject with different gender?

Source: Colah

Solution - Long Memory and Short Memory



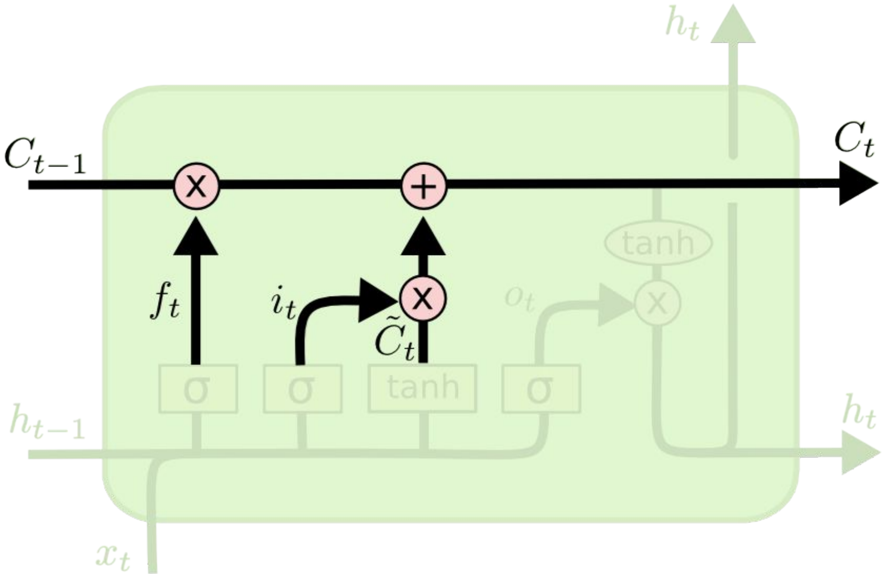
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$

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

Input Gate Layer

Source: Colah

Solution - Long Memory and Short Memory

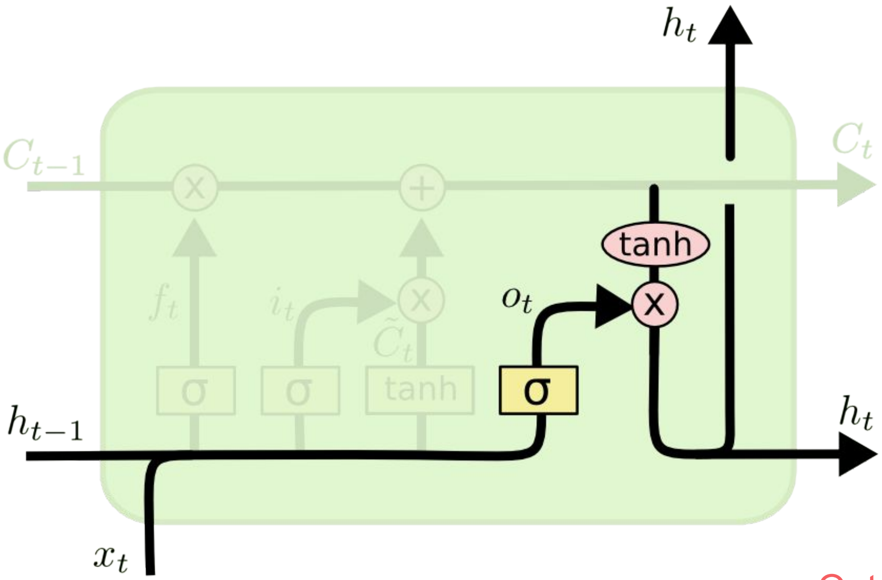


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Combine to make current state

Source: Colah

Solution - Long Memory and Short Memory



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

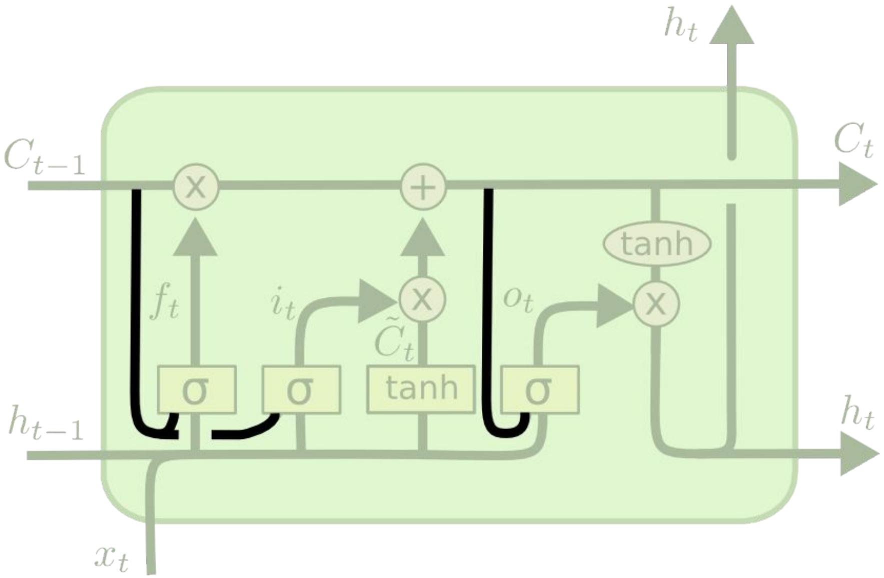
$$h_t = o_t * \tanh\left(C_t\right)$$

Output Gate Layer

Why current input in the state? So that, things like plurality of subject can be determined.

Source: Colah

Even better LSTM



$$f_t = \sigma\left(W_f \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \;+\; b_f\right)$$
$$i_t = \sigma\left(W_i \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \;+\; b_i\right)$$
$$o_t = \sigma\left(W_o \cdot [\boldsymbol{C_t}, h_{t-1}, x_t] \;+\; b_o\right)$$

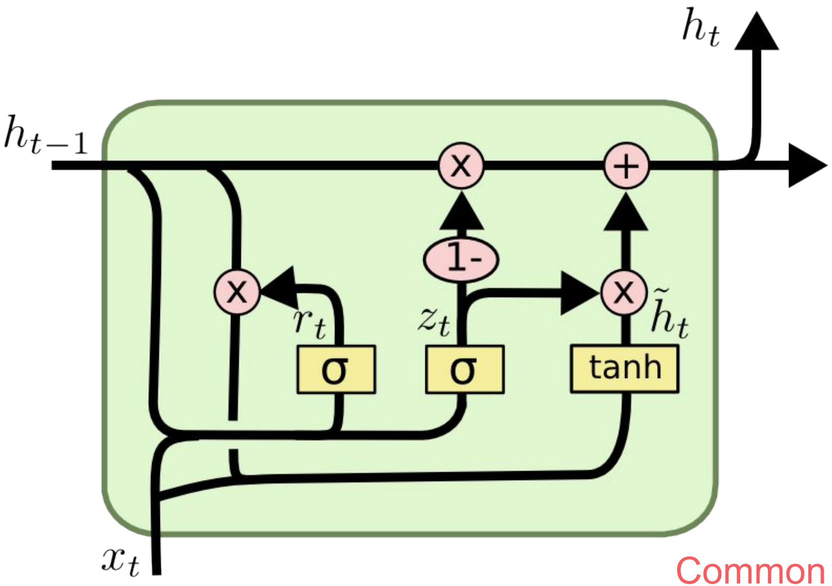
LSTM with “peephole connections”

Gate layers look at the cell state.

Gers & Schmidhuber (2000)

Source: Colah

GRU - Gated Recurrent Unit



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

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

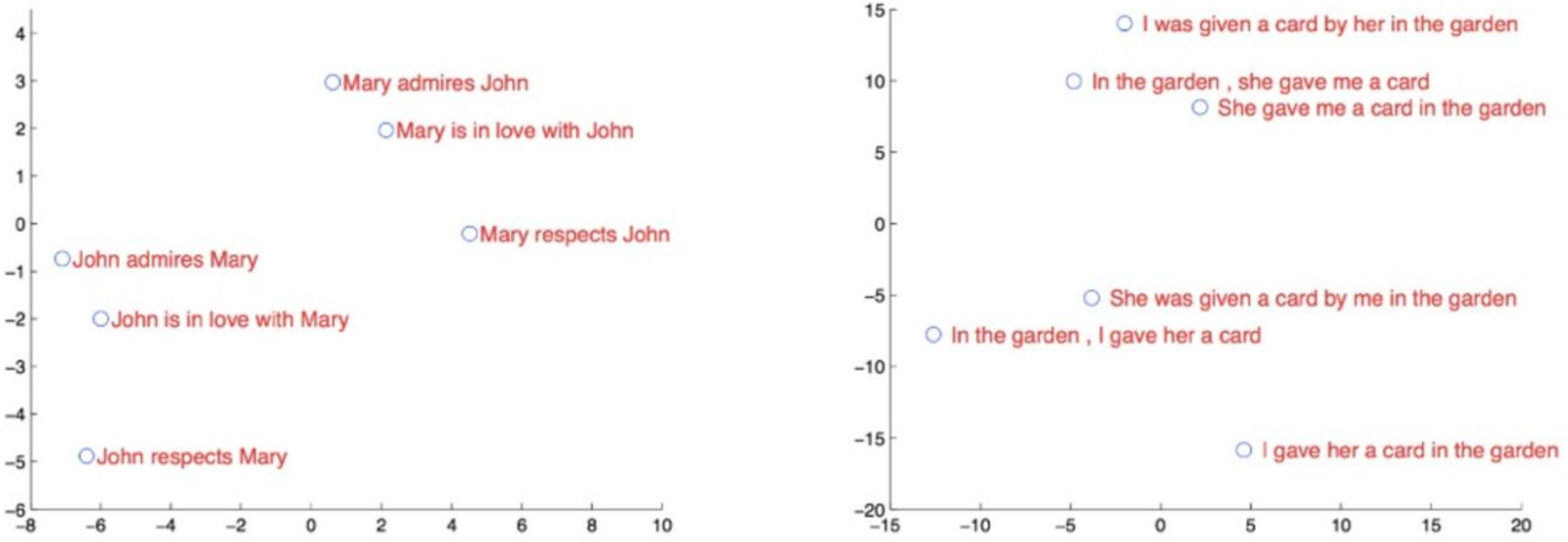
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Common cell state and hidden state

Combines the forget and input gates into a single update gate Cho, et al. (2014)

Source: Colah

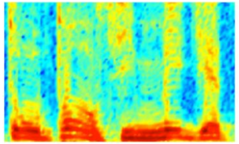
Learned Representation



2D Visualization of ‘vectors’ learned for sentences. Similar sentences are close together in ‘vector’ space. [Sutskever et al, 2014](#)

Word Vectors

AUDIO | IMAGES | TEXT



Audio Spectrogram

DENSE

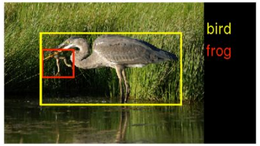


Image pixels

DENSE

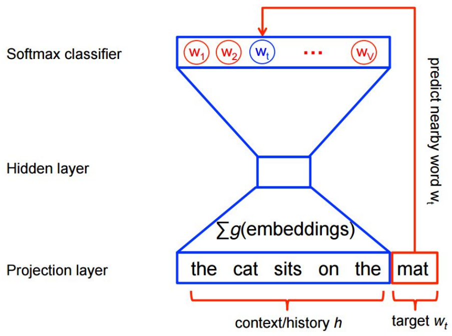
Word, context, or document vectors

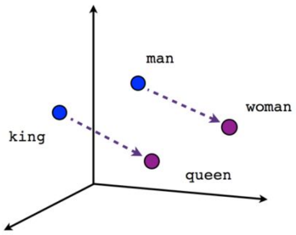
SPARSE

- To get dense representation of words.
- These dense vectors encode the meaning of the word based on context.
- Because --

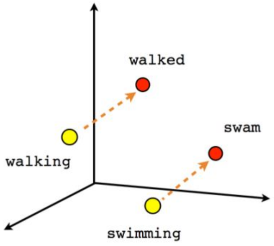
  You shall know a word by the company it keeps - J. R. Firth, 1957
- Usually dimensions of these vectors are 100 to 1000 (much smaller than 100K)

Word Vectors

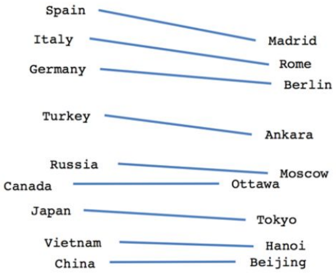




Male-Female



Verb tense

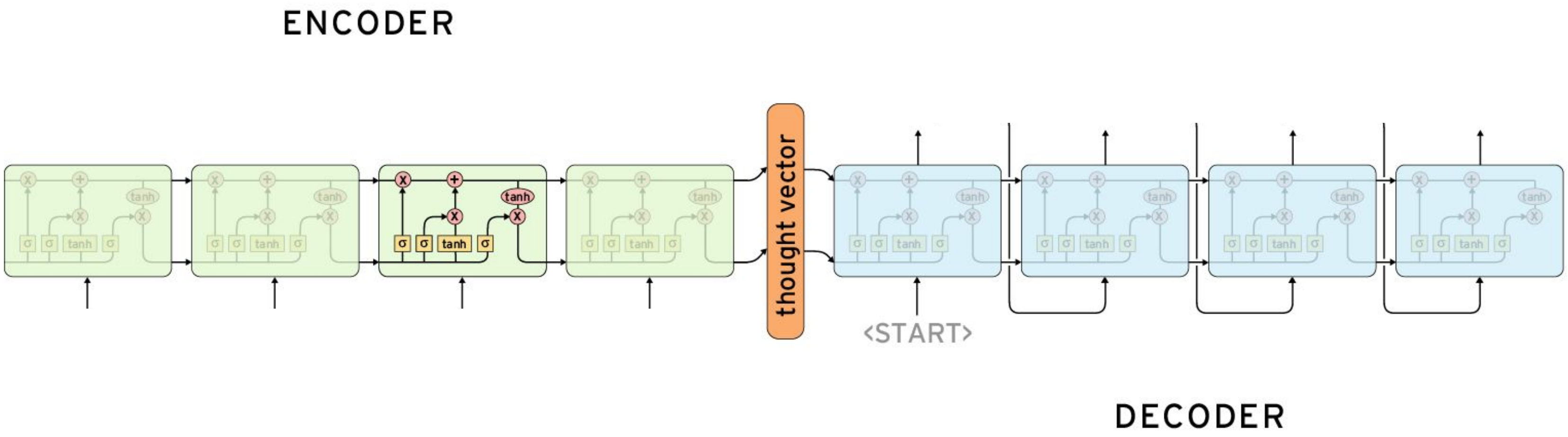


Country-Capital

- CBOW - Predict the target word given the source context words.
- Skip-Gram - Predict the source context-words given the target words.
- These dense vectors encode the meaning of the word based on context.
- Because --

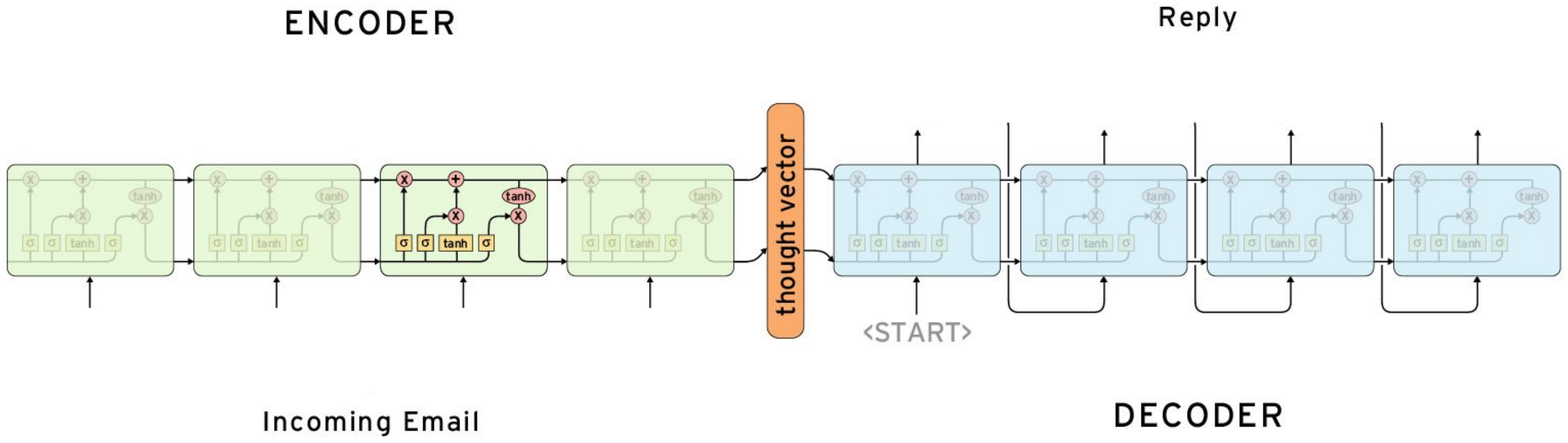
  You shall know a word by the company it keeps - J. R. Firth, 1957
- Usually dimensions of these vectors are 100 to 1000 (much smaller than 100K)

Sequence to Sequence



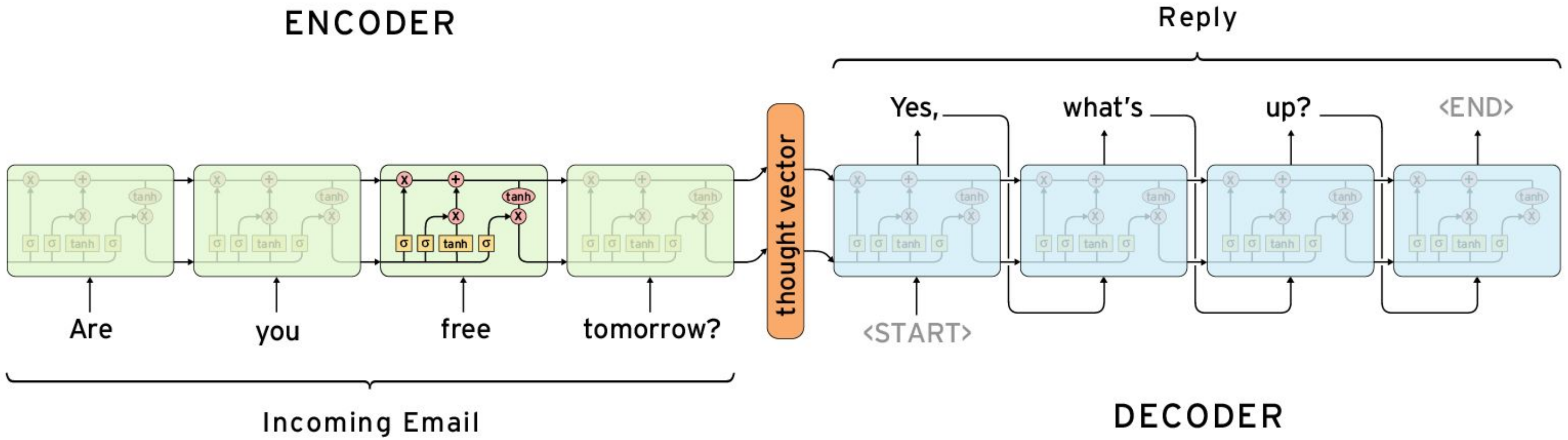
Source: Colah

Sequence to Sequence



Source: Colah

Sequence to Sequence



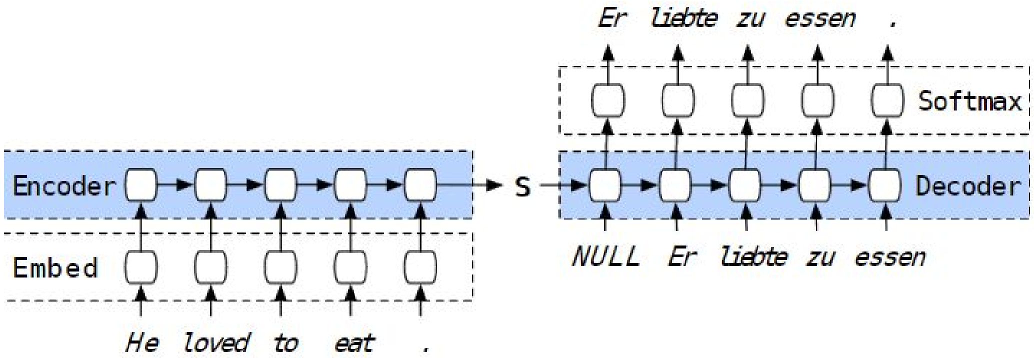
Source: Colah

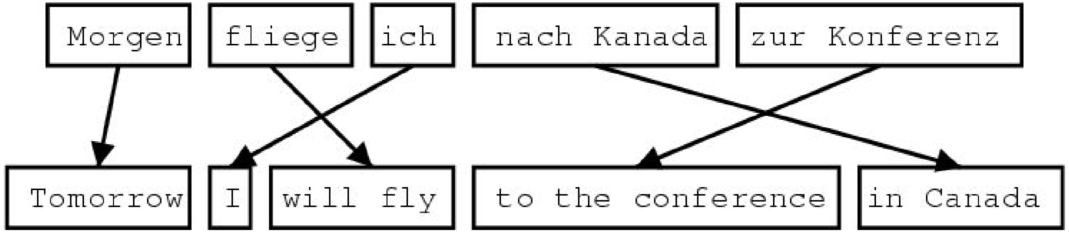
Applications - Plenty

1. Dialog generation
2. Machine translation
3. Image captioning
4. Paraphrasing
5. Speech Recognition
6. Handwriting Recognition

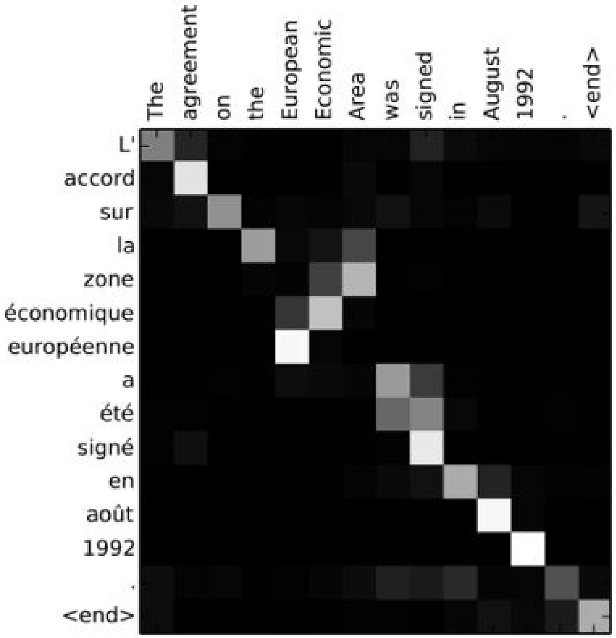
Reddit Comments	- Demo
US Elections Dialog agent	- Demo
Image captioning	- Demo
Handwriting Recognition	- Demo

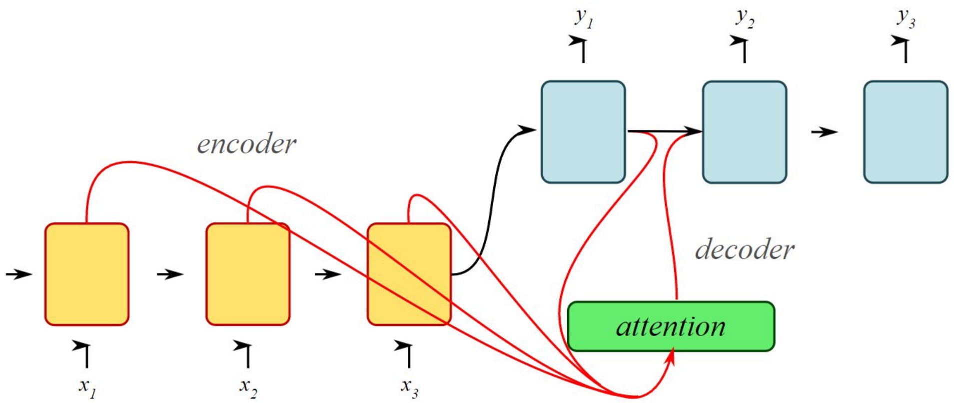
Neural Machine Translation - Alignment





NMT - Attention

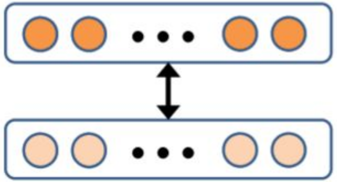




Single-modal learning

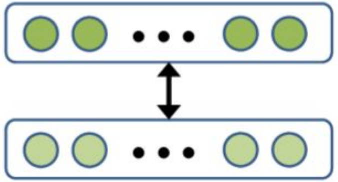
Images

- Classification
- Segmentation
- Detection

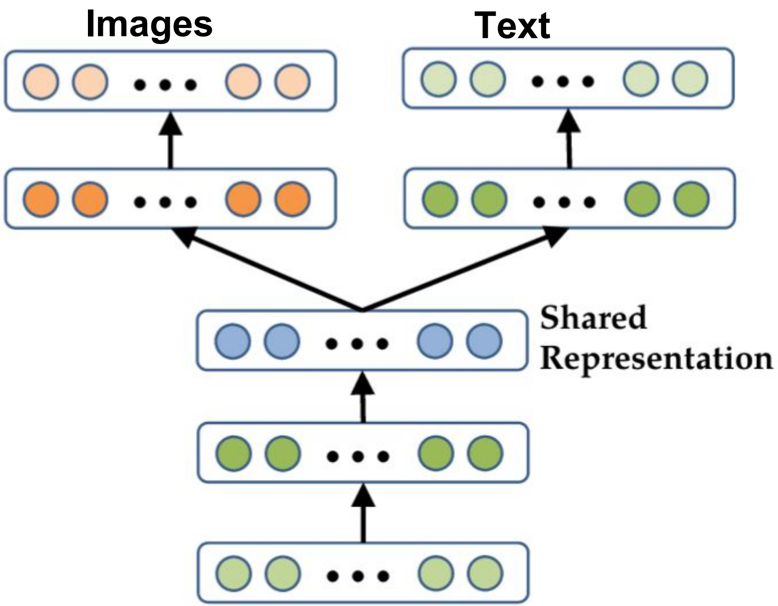


Text

- Parsing
- Translation
- Question Answering



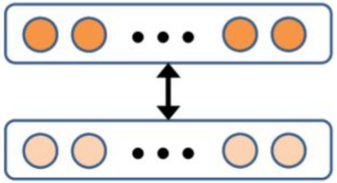
Multi-modal Learning



Single-modal learning

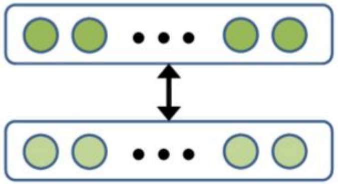
Images

- Classification
- Segmentation
- Detection



Text

- Parsing
- Translation
- Question Answering



Multi-modal Learning

1. Visual Question Answering
2. Image Captioning
3. Video summarization (Images + Audio)

Image Captioning - Show and Tell

- Given an image, output possible sentences to describe the image.
- The sentence could have varying length.
- Use CNN for image modelling and RNN for language

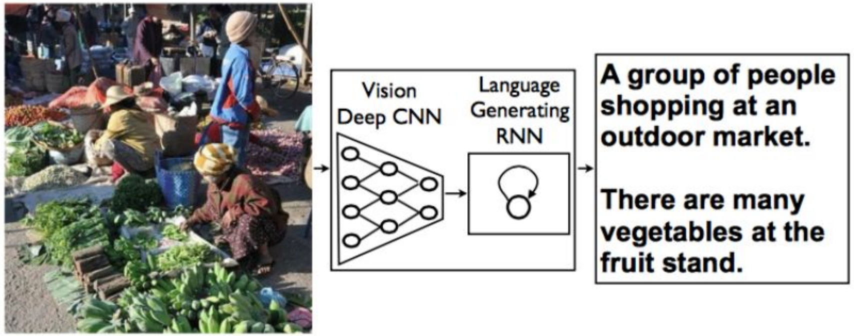
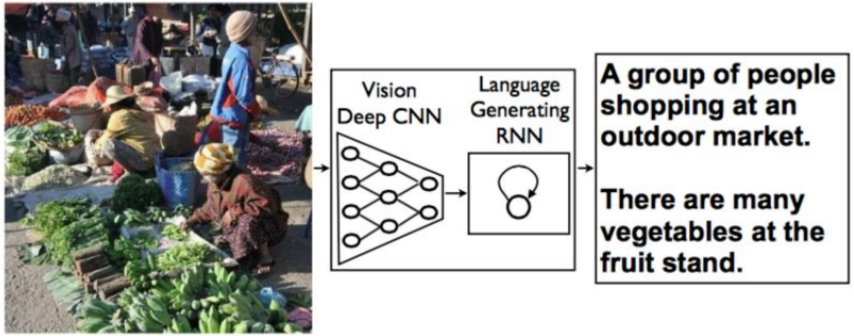


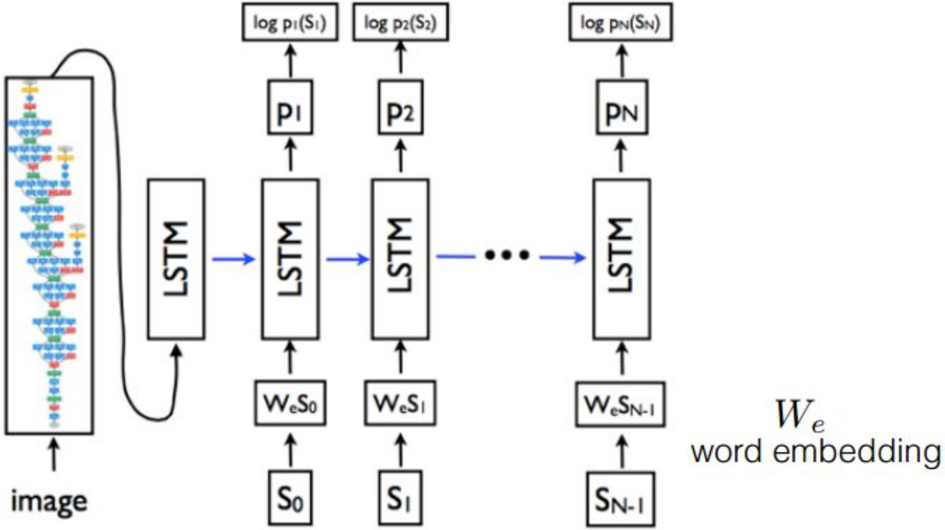
Image Captioning - Show and Tell

Training: BPTT $log\ p(S|I) = \sum_{t=1} log\ p(S_t|I, S_0 \ldots S_{t-1})$

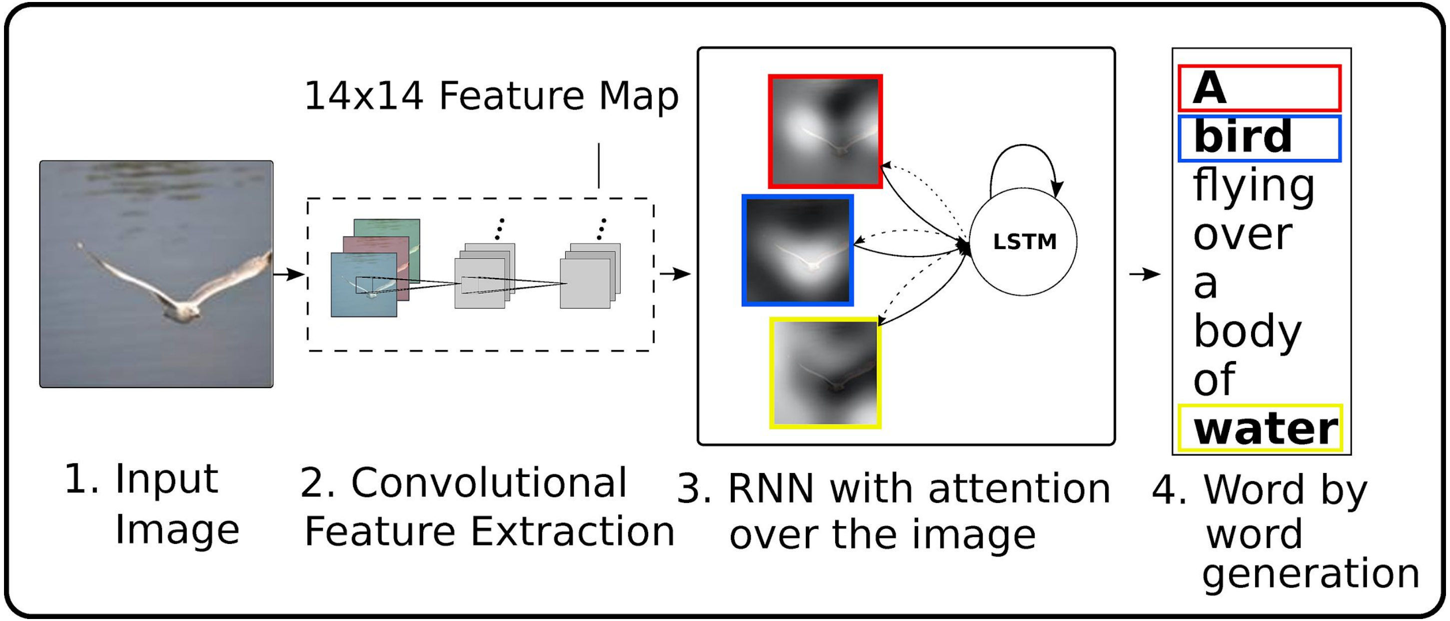
$$loss(S, I) = -\sum_{t=1}^{N} p_t(S_t)$$

- Given an image, output possible sentences to describe the image.
- The sentence could have varying length.
- Use CNN for image modelling and RNN for language





Visual Attention, Show, attend & Tell paper



Let every step of an RNN pick information to look at from some larger collection of information

Attention Model in action

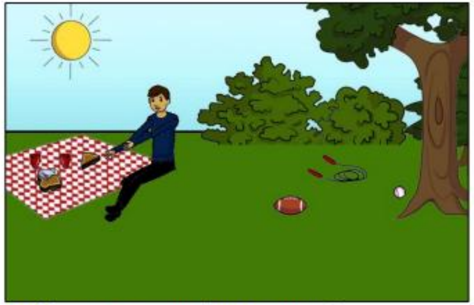
Visual Question Answering

VQA is a new dataset containing open-ended questions about images. These questions require an understanding of vision, language and commonsense knowledge to answer.

- Fine grained recognition - “what kind of cheese?”
- Object Detection - “How many bikes?”
- Activity Recognition - “Is this man crying?”
- Reasoning - “Is this pizza vegetarian?”
- Common sense - “Does this person have 20/20 vision?”

What color are her eyes?
What is the mustache made of?

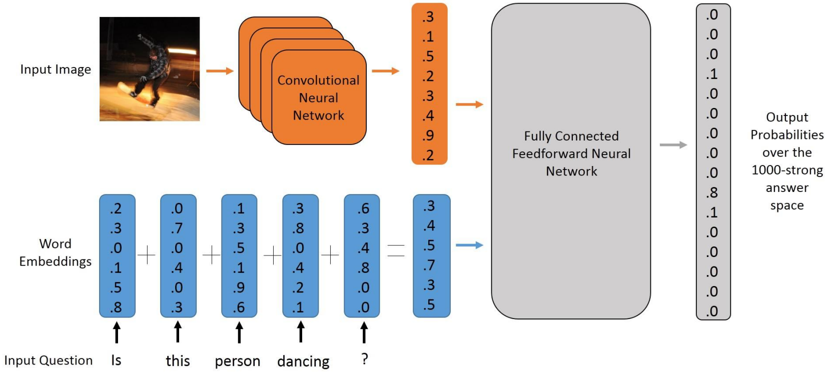
How many slices of pizza are there?
Is this a vegetarian pizza?

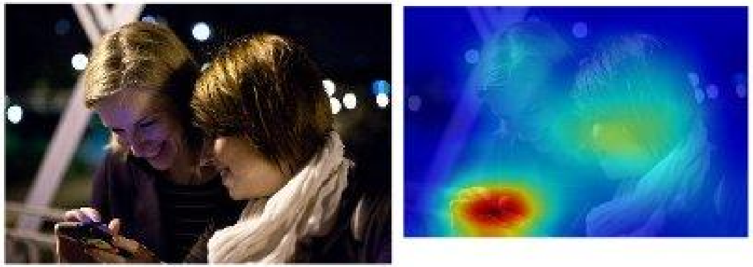


Is this person expecting company?
What is just under the tree?

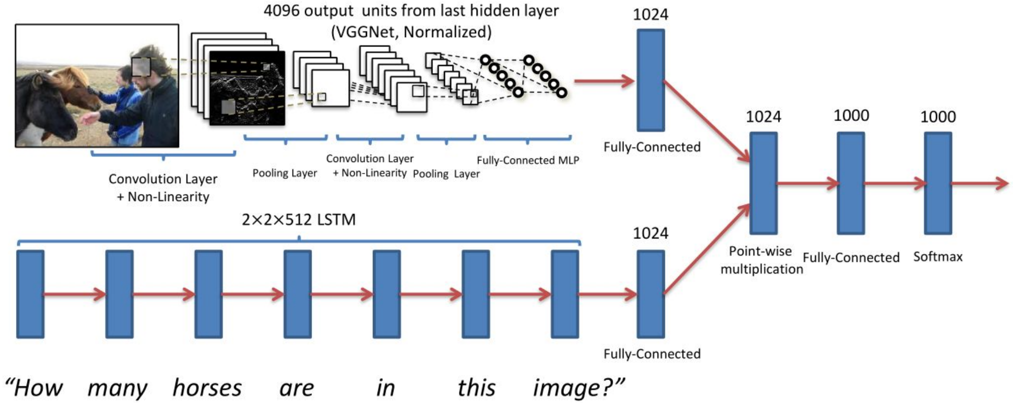
Does it appear to be rainy?
Does this person have 20/20 vision?

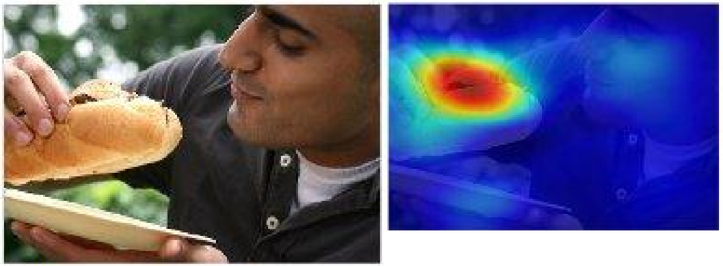
Visual Question Answering





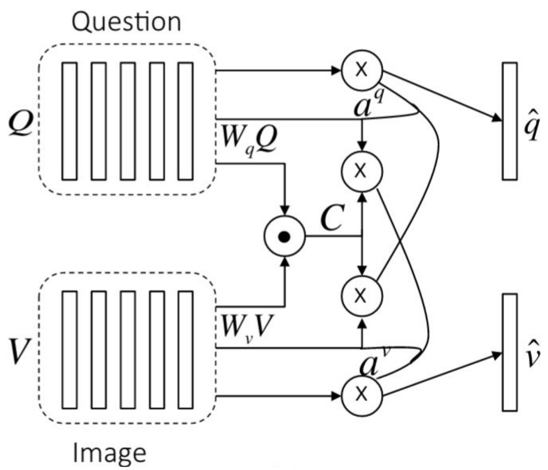
Question: What are they doing?
Prediction: texting (score: 12.02=3.78 [image] + 8.24 [word])
Word importance: doing(7.01) are(1.05) they(0.49) what(-0.3)

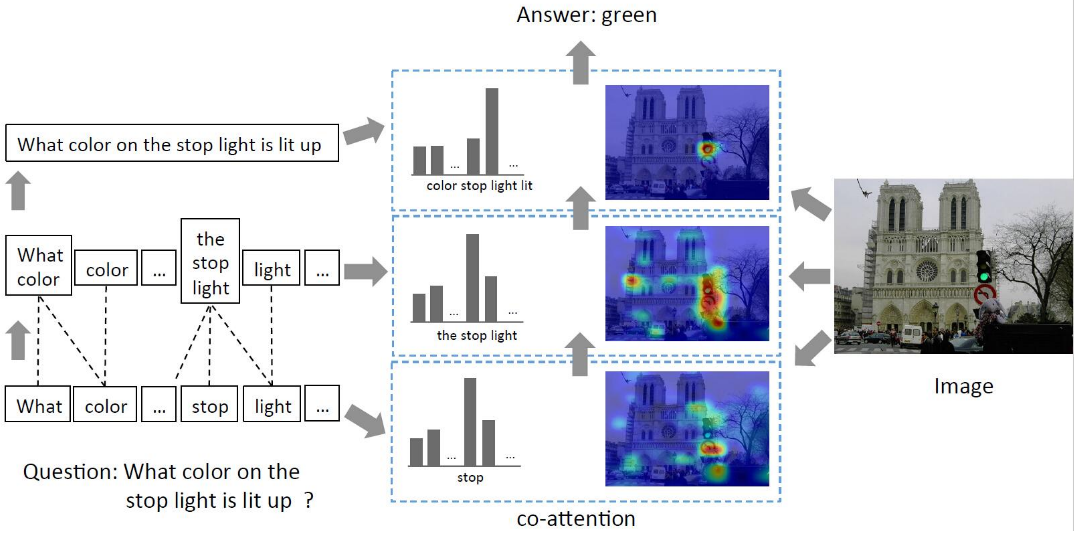




Question: What is he eating?
Prediction: hot dog (score: 13.01=5.02 [image] + 7.99 [word])
Word importance: eating(4.12) what(2.81) is(0.74) he(0.30)

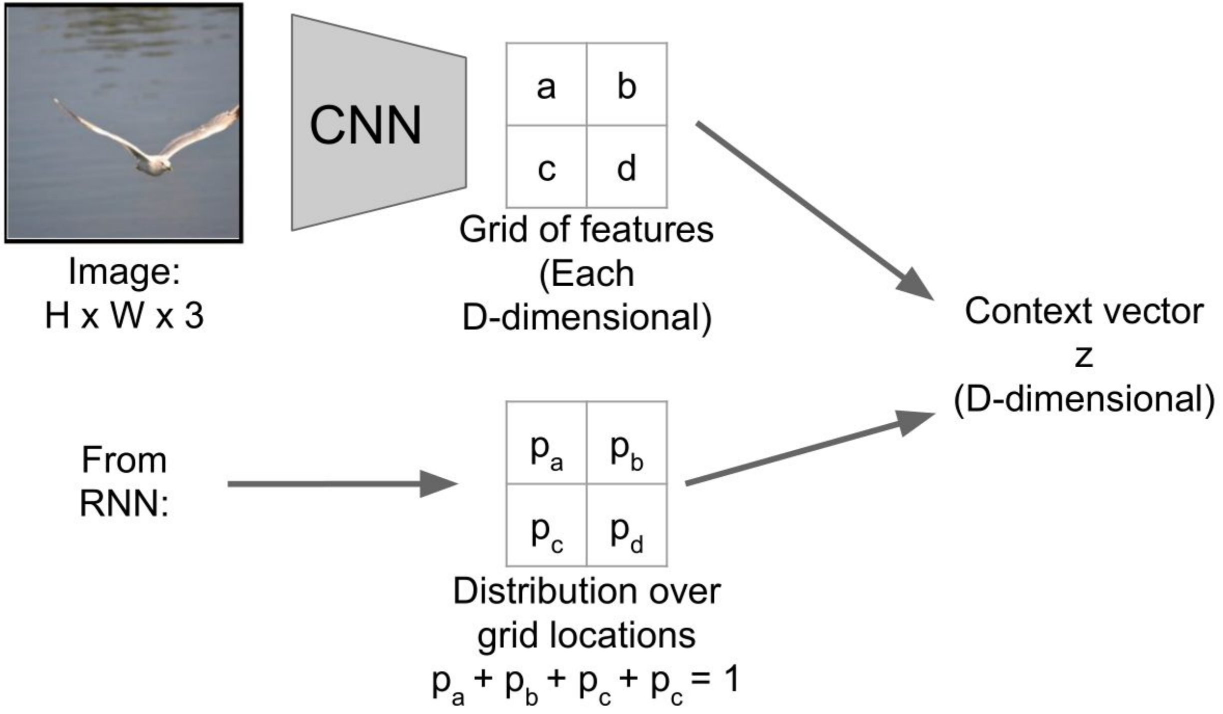
Visual Question Answering - Attention





Source: Jiasen

Soft Attention



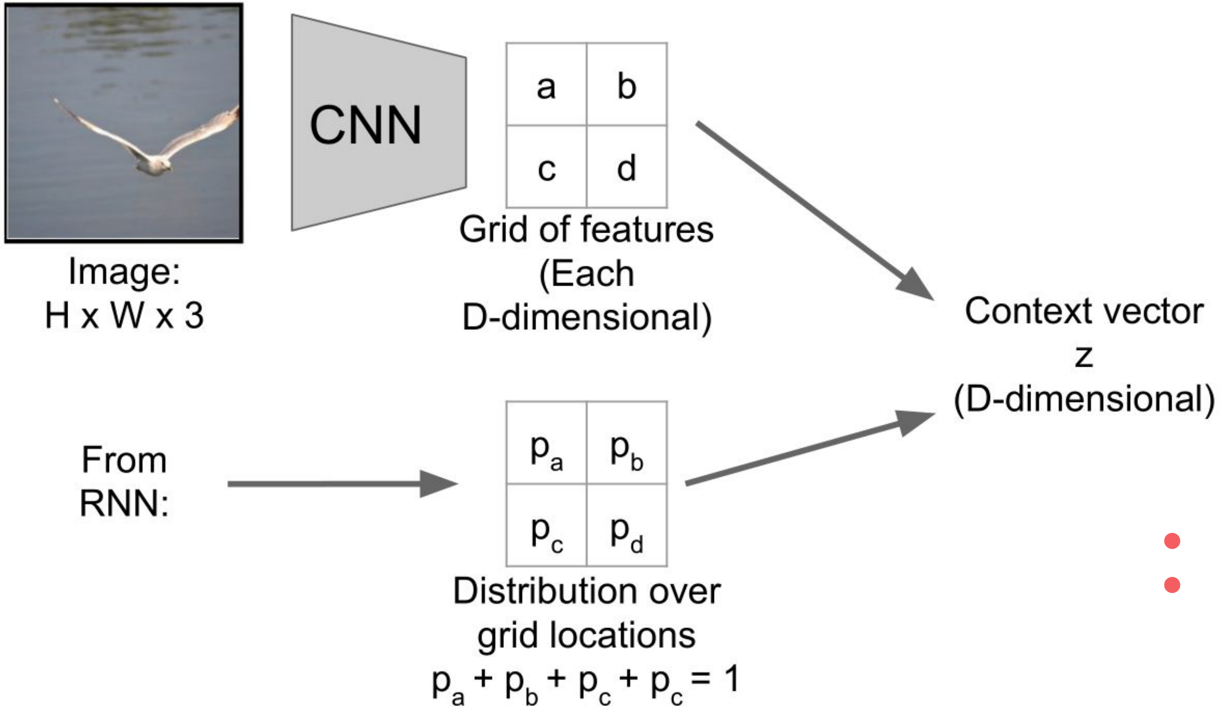
Summarize ALL locations

$z = p_a a + p_b b + p_c c + p_d d$

Derivative dz/dp is nice!

Train with gradient descent

Soft Attention



Summarize ALL locations
$z = p_a a + p_b b + p_c c + p_d d$

Derivative dz/dp is nice!
Train with gradient descent

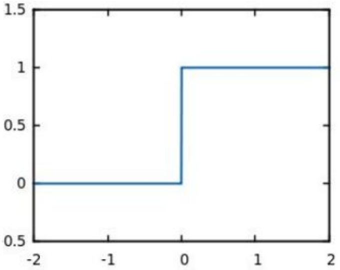
- Still uses the whole input
- Constrained to fix grid

Hard Attention

Input image:
H x W x 3

Cropped and
rescaled image:
X x Y x 3

Box Coordinates:
(xc, yc, w, h)



Gradient is 0 almost everywhere
Gradient is undefined at x = 0

Hard attention:
Sample a subset
of the input

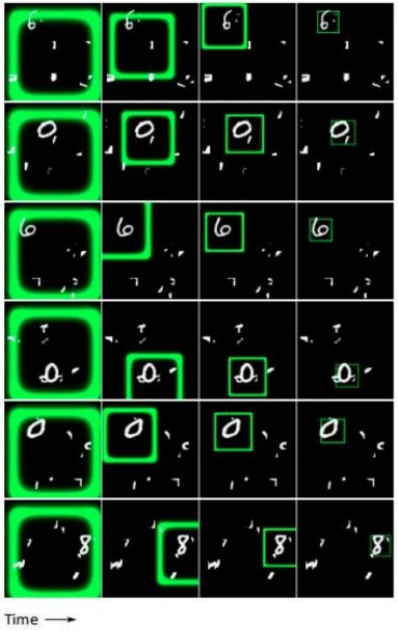
Not a differentiable function !

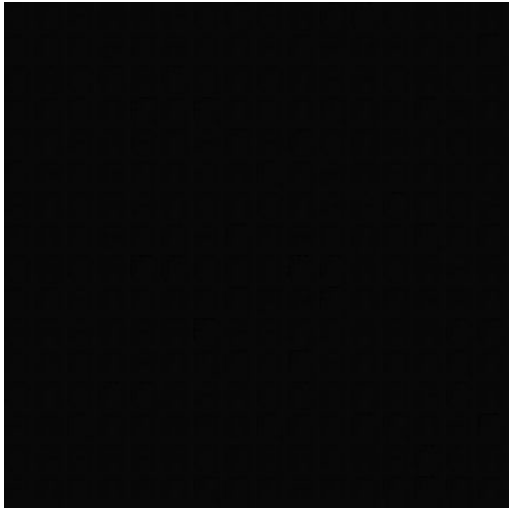
Can’t train with backprop :(

need [reinforcement learning]

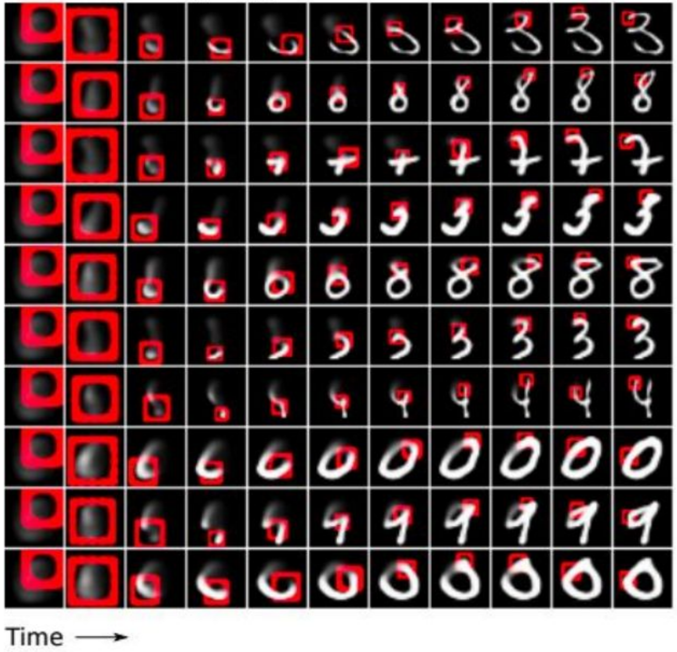
Hard Attention - Use

Classify images by attending to arbitrary regions of the *input*



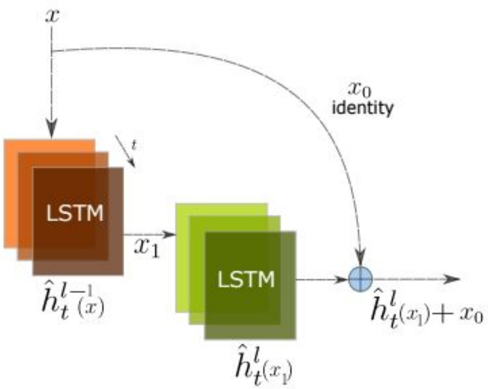


Generate images by attending to arbitrary regions of the *output*



[DRAW: A Recurrent Neural Network, DeepMind]()

Neural Paraphrase Generation



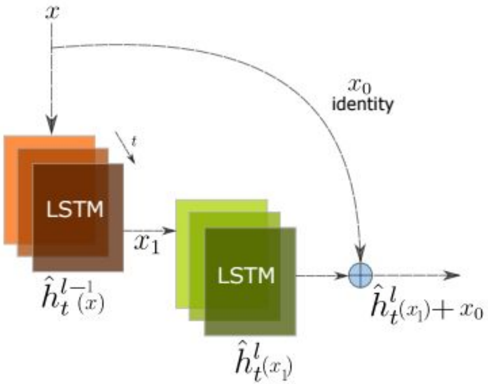
a small kitten is sitting in a bowl
a cat is curled up in a bowl
a cat that is sitting on a bowl

an old couple at the beach during the day
two people sitting on dock looking at the ocean
a couple standing on top of a sandy beach

a little baby is sitting on a huge motorcycle
a little boy sitting alone on a motorcycle
a baby sitting on top of a motorcycle

Source: [Yours kindly]

Neural Paraphrase Generation

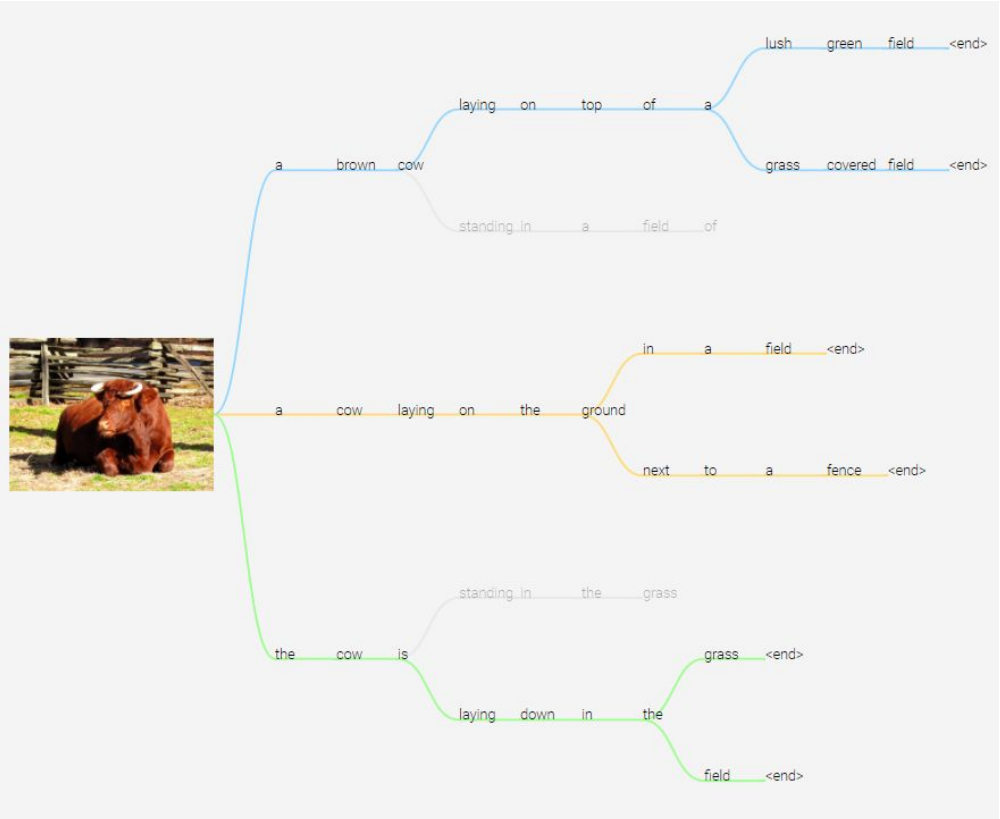


a small kitten is sitting in a bowl	**Source**	south eastern
a cat is curled up in a bowl	**Reference**	the eastern part
a cat that is sitting on a bowl	**Generated**	south east
an old couple at the beach during the day	**Source**	organized
two people sitting on dock looking at the ocean	**Reference**	managed
a couple standing on top of a sandy beach	**Generated**	arranged
a little baby is sitting on a huge motorcycle	**Source**	counselling
a little boy sitting alone on a motorcycle	**Reference**	be kept informed
a baby sitting on top of a motorcycle	**Generated**	consultations

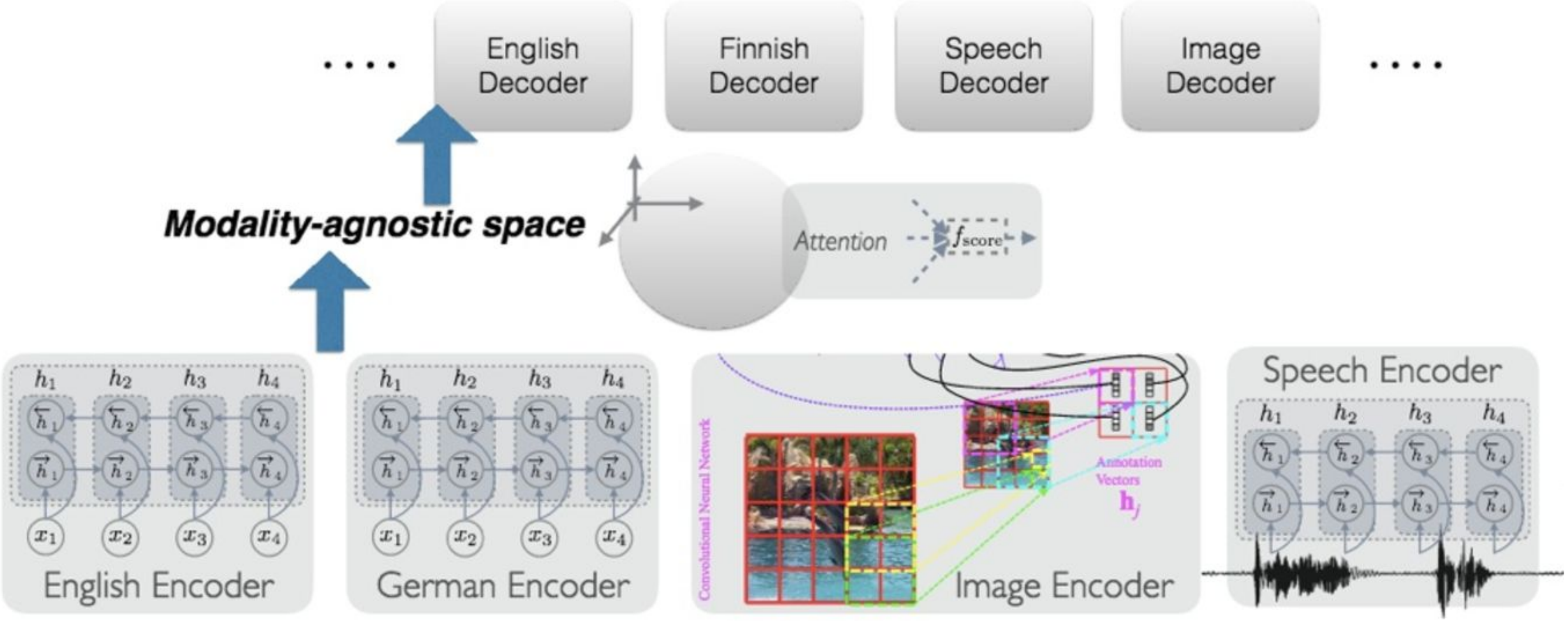
Source: Yours kindly

Beam Search (encourage diversity)

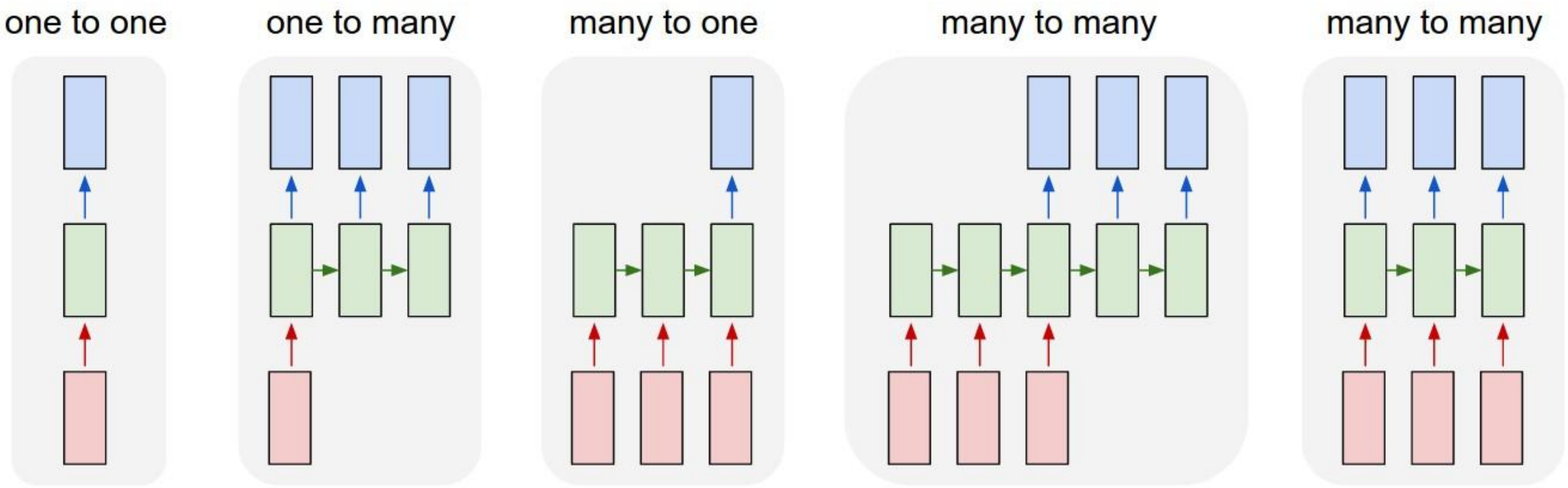
- Greedy Search is efficient but not optimal.
- Maybe the second best word is a good solution given rest of the words that will become part of the sequence.
- Beam Search - Maintain ‘K’ hypotheses at a time.
- Expand each hypothesis.
- Pick top-K hypotheses at each time step.
- Demo



Sequence to Sequence - Modality agnostic

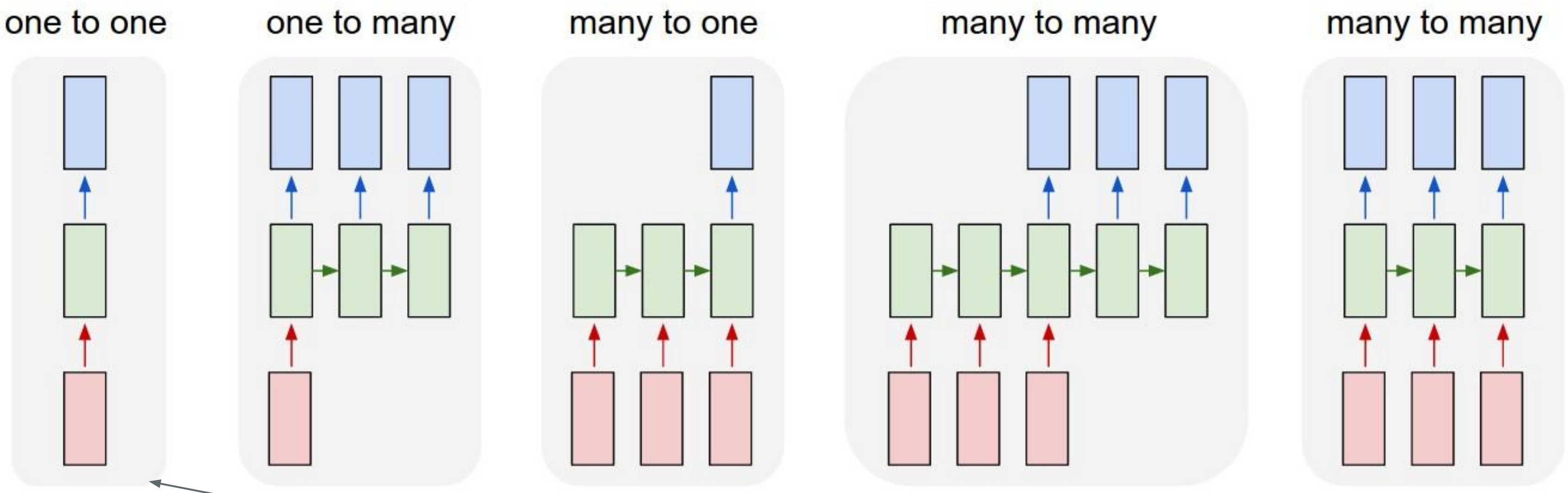


Recurrence in learning



Source: Karpathy

Recurrent Neural Networks - A recap



Vanilla Neural Networks

Source: Karpathy

Recurrent Neural Networks - A recap

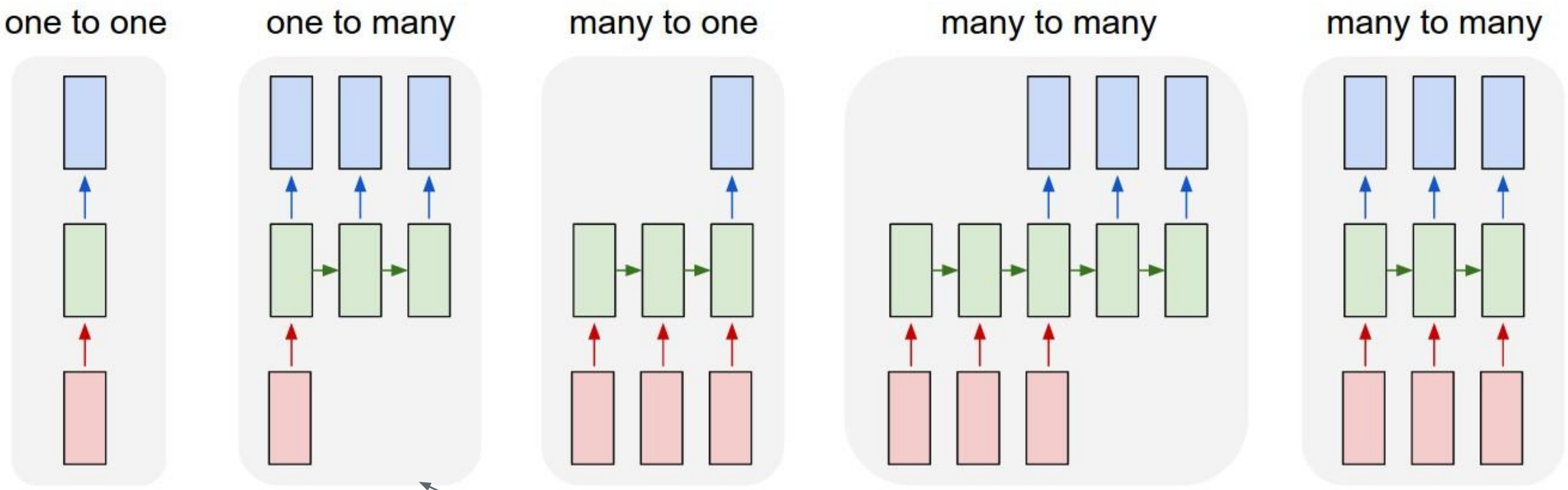
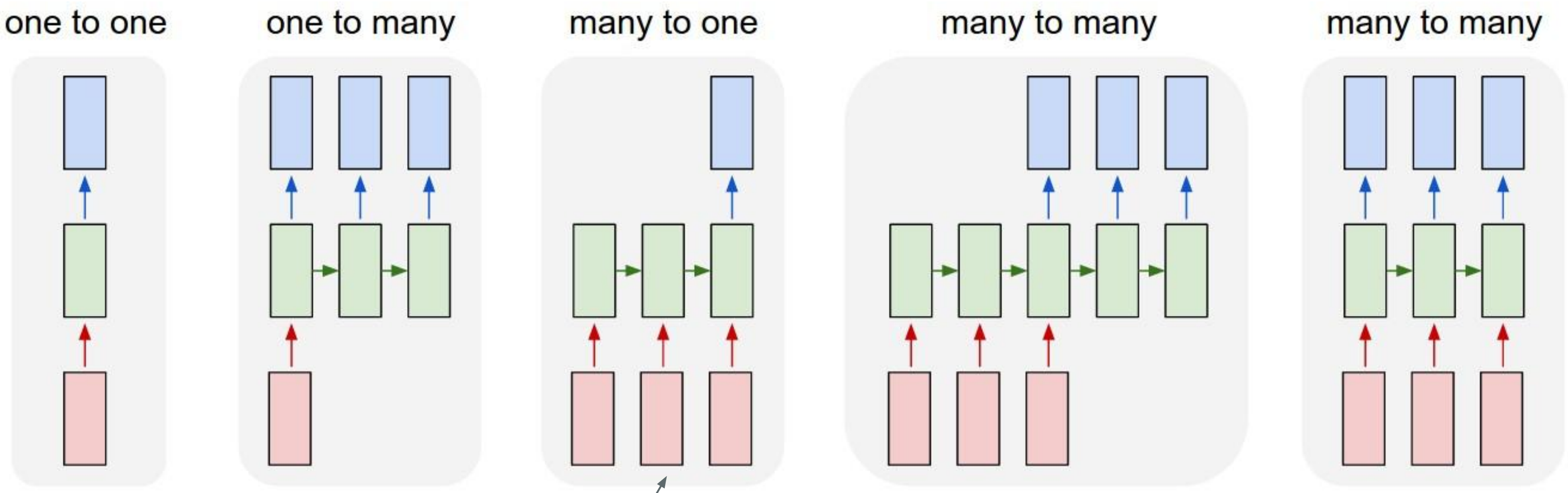


Image Captioning -
Sequence to Words

Source: Karpathy

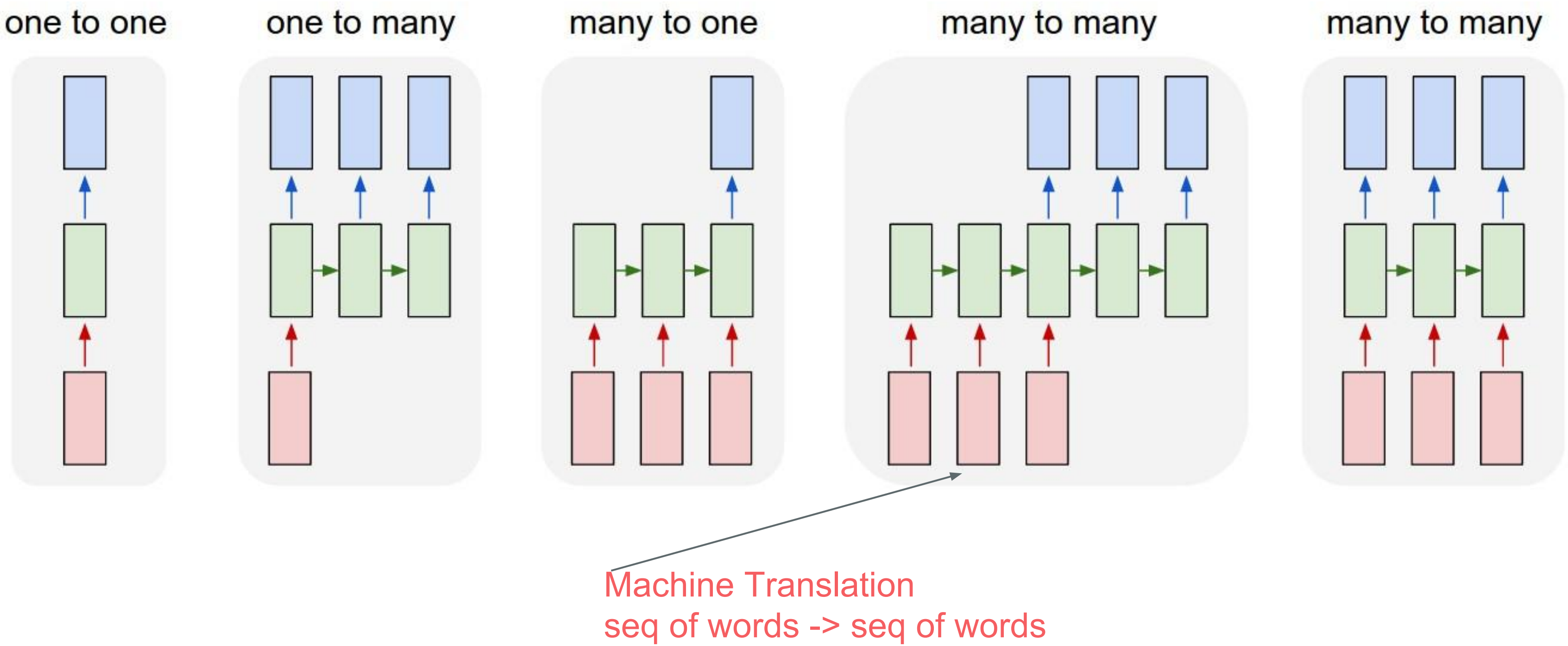
Recurrent Neural Networks - A recap



Sentiment Classification
sequence of words -> sentiment

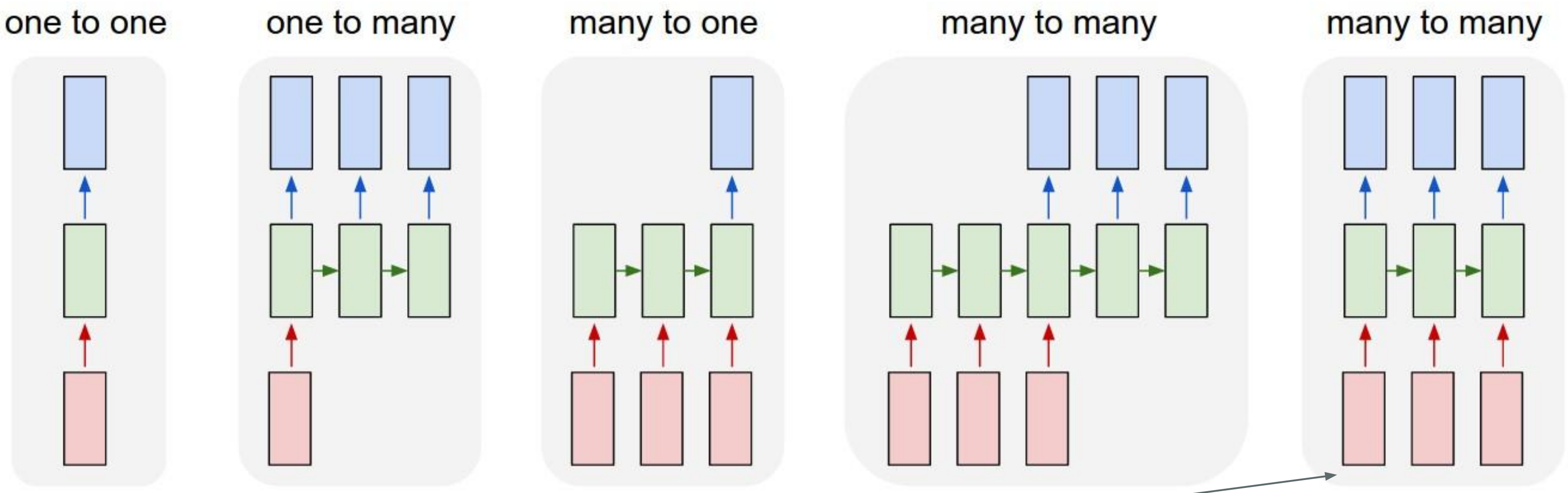
Source: Karpathy

Recurrent Neural Networks - A recap



Source: Karpathy

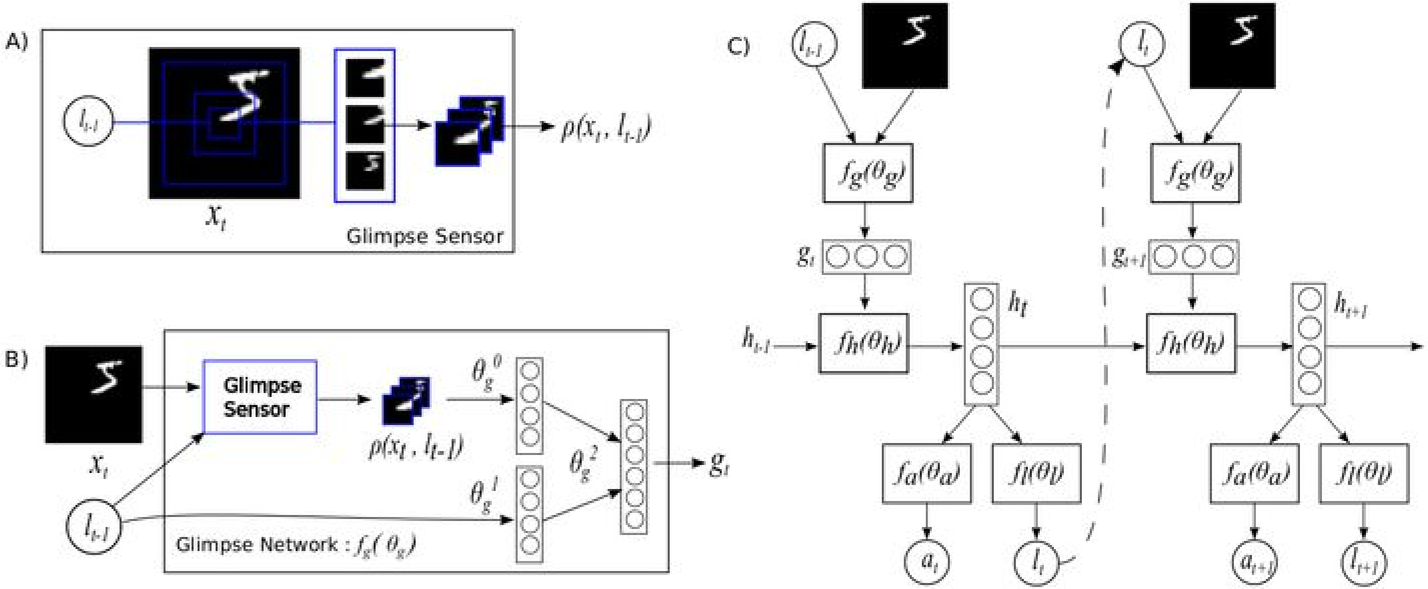
Recurrent Neural Networks - A recap



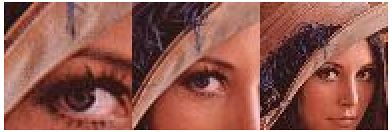
Video classification (frame level)
VQA - ??? More on this later

Source: Karpathy

RMVA Recurrent Models of Visual Attention



- Glimpse sensor : bandwidth limited sensor of the input image. As an example, if the input image is of size 28x28 (height x width), the RAM may only be able to sense an area of size 8x8 at any given time-step, called glimpses

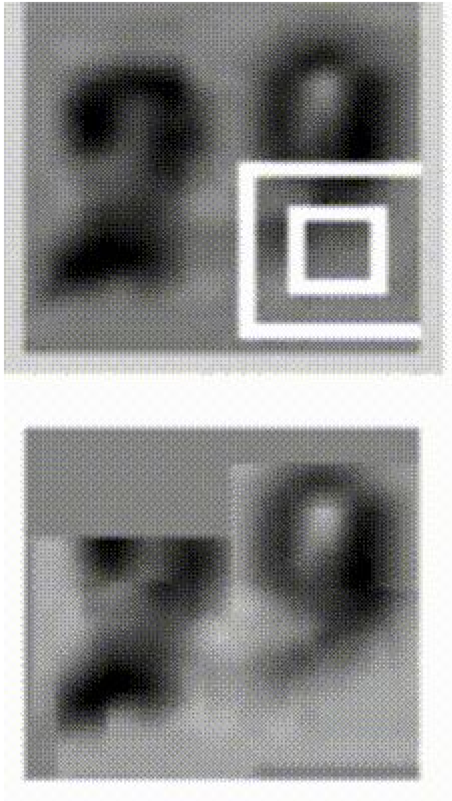


Unreasonable effectiveness of ~~RNN~~ LSTM Images -

Even traditional areas where CNN has done excellent are being improved by use of RNN.

- Reads number left to right (steps)

Work by DeepMind
http://arxiv.org/abs/1412.7755



Unreasonable effectiveness of ~~RNN~~ LSTM Literature -

- Char level LSTM
- Trained on all works of Shakespeare
- 3-layer RNN with 512 hidden nodes
- Learns perfect spelling !

```
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.

Clown:
Come, sir, I will make did behold your worship.

VIOLA:
I'll drink it.
```

Unreasonable effectiveness of ~~RNN~~ LSTM

Math & Latex -

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. □

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

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

is an open subset of X. Thus U is affine. This is a continuous map of X is the inverse, the groupoid scheme S.

Proof. See discussion of sheaves of sets. □

The result for prove any open covering follows from the less of Example ??. It may replace S by $X_{spaces,\acute{e}tale}$ which gives an open subspace of X and T equal to S_{Zar}, see Descent, Lemma ??. Namely, by Lemma ?? we see that R is geometrically regular over S.

Lemma 0.1. *Assume (3) and (3) by the construction in the description.*

Suppose $X = \lim |X|$ (by the formal open covering X and a single map $\underline{Proj}_X(\mathcal{A}) = \mathrm{Spec}(B)$ over U compatible with the complex

$$Set(\mathcal{A}) = \Gamma(X, \mathcal{O}_{X,\mathcal{O}_X}).$$

When in this case of to show that $\mathcal{Q} \to \mathcal{C}_{Z/X}$ is stable under the following result in the second conditions of (1), and (3). This finishes the proof. By Definition ?? (without element is when the closed subschemes are catenary. If T is surjective we may assume that T is connected with residue fields of S. Moreover there exists a closed subspace $Z \subset X$ of X where U in X' is proper (some defining as a closed subset of the uniqueness it suffices to check the fact that the following theorem

(1) *f is locally of finite type. Since $S = \mathrm{Spec}(R)$ and $Y = \mathrm{Spec}(R)$.*

Proof. This is form all sheaves of sheaves on X. But given a scheme U and a surjective étale morphism $U \to X$. Let $U \cap U = \coprod_{i=1,\ldots,n} U_i$ be the scheme X over S at the schemes $X_i \to X$ and $U = \lim_i X_i$. □

The following lemma surjective restrocomposes of this implies that $\mathcal{F}_{x_0} = \mathcal{F}_{x_0} = \mathcal{F}_{X,\ldots,0}$.

Lemma 0.2. *Let X be a locally Noetherian scheme over S, $E = \mathcal{F}_{X/S}$. Set $\mathcal{I} = \mathcal{J}_1 \subset \mathcal{I}'_n$. Since $\mathcal{I}^n \subset \mathcal{I}^n$ are nonzero over $i_0 \leq \mathfrak{p}$ is a subset of $\mathcal{J}_{n,0} \circ \overline{A}_2$ works.*

Lemma 0.3. *In Situation ??. Hence we may assume $\mathfrak{q}' = 0$.*

Proof. We will use the property we see that $\mathfrak{p}$ is the mext functor (??). On the other hand, by Lemma ?? we see that

$$D(\mathcal{O}_{X'}) = \mathcal{O}_X(D)$$

where K is an F-algebra where δ_{n+1} is a scheme over S. □

Unreasonable effectiveness of ~~RNN~~ LSTM

Math & Latex & Drawing

Proof. Omitted. $\square$

Lemma 0.1. *Let $\mathcal{C}$ be a set of the construction.*

Let $\mathcal{C}$ be a gerber covering. Let $\mathcal{F}$ be a quasi-coherent sheaves of $\mathcal{O}$-modules. We have to show that

$$\mathcal{O}_{\mathcal{O}_X} = \mathcal{O}_X(\mathcal{L})$$

.

Proof. This is an algebraic space with the composition of sheaves $\mathcal{F}$ on $X_{étale}$ we have

$$\mathcal{O}_X(\mathcal{F}) = \{morph_1 \times_{\mathcal{O}_X} (\mathcal{G}, \mathcal{F})\}$$

where $\mathcal{G}$ defines an isomorphism $\mathcal{F} \to \mathcal{F}$ of $\mathcal{O}$-modules. $\square$

Lemma 0.2. *This is an integer $\mathcal{Z}$ is injective.*

Proof. See Spaces, Lemma ??. $\square$

Lemma 0.3. *Let S be a scheme. Let X be a scheme and X is an affine open covering. Let $\mathcal{U} \subset \mathcal{X}$ be a canonical and locally of finite type. Let X be a scheme. Let X be a scheme which is equal to the formal complex.*

The following to the construction of the lemma follows.

Let X be a scheme. Let X be a scheme covering. Let

$$b : X \to Y' \to Y \to Y \to Y' \times_X Y \to X.$$

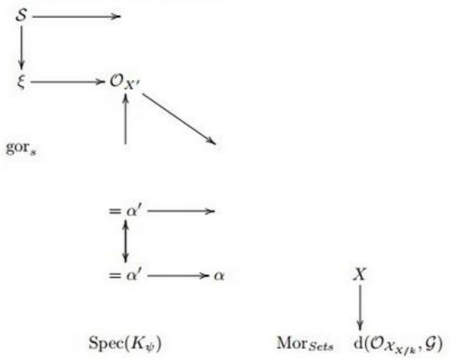
be a morphism of algebraic spaces over S and Y.

Proof. Let X be a nonzero scheme of X. Let X be an algebraic space. Let $\mathcal{F}$ be a quasi-coherent sheaf of $\mathcal{O}_X$-modules. The following are equivalent

1. (1) $\mathcal{F}$ is an algebraic space over S.
2. (2) If X is an affine open covering.

Consider a common structure on X and X the functor $\mathcal{O}_X(U)$ which is locally of finite type. $\square$

This since $\mathcal{F} \in \mathcal{F}$ and $x \in \mathcal{G}$ the diagram



is a limit. Then $\mathcal{G}$ is a finite type and assume S is a flat and $\mathcal{F}$ and $\mathcal{G}$ is a finite type f_*. This is of finite type diagrams, and

- the composition of $\mathcal{G}$ is a regular sequence,
- $\mathcal{O}_{X'}$ is a sheaf of rings.

$\square$

Proof. We have see that $X = \mathrm{Spec}(R)$ and $\mathcal{F}$ is a finite type representable by algebraic space. The property $\mathcal{F}$ is a finite morphism of algebraic stacks. Then the cohomology of X is an open neighbourhood of U. $\square$

Proof. This is clear that $\mathcal{G}$ is a finite presentation, see Lemmas ??.
A *reduced above* we conclude that U is an open covering of $\mathcal{C}$. The functor $\mathcal{F}$ is a "field

$$\mathcal{O}_{X,x} \longrightarrow \mathcal{F}_{\overline{x}} \ \text{-1}(\mathcal{O}_{X_{étale}}) \longrightarrow \mathcal{O}_{X_\ell}^{-1}\mathcal{O}_{X_\lambda}(\mathcal{O}_{X_\eta}^{\overline{v}})$$

is an isomorphism of covering of $\mathcal{O}_{X_i}$. If $\mathcal{F}$ is the unique element of $\mathcal{F}$ such that X is an isomorphism.
The property $\mathcal{F}$ is a disjoint union of Proposition ?? and we can filtered set of presentations of a scheme $\mathcal{O}_X$-algebra with $\mathcal{F}$ are opens of finite type over S.
If $\mathcal{F}$ is a scheme theoretic image points. $\square$

If $\mathcal{F}$ is a finite direct sum $\mathcal{O}_{X_\lambda}$ is a closed immersion, see Lemma ??. This is a sequence of $\mathcal{F}$ is a similar morphism.

Unreasonable effectiveness of ~~RNN~~ LSTM

Linux
Source Code

```
/*
 * 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;
    }
  }
```

Unreasonable effectiveness of ~~RNN~~ LSTM Bible !

References

1. http://karpathy.github.io/2015/05/21/rnn-effectiveness/
2. http://colah.github.io/posts/2015-08-Understanding-LSTMs/
3. http://iamtrask.github.io/2015/11/15/anyone-can-code-lstm/
4. Neural Turing Machines, Graves et al http://arxiv.org/pdf/1410.5401v2.pdf
5. www.technologyreview.com/view/532156/googles-secretive-deepmind-startup-unveils-a-neural-turing-machine/
6. Recurrent Models of Visual Attention, Mnih et al, http://papers.nips.cc/paper/5542-recurrent-models-of-visual-attention.pdf
7. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention, Kelvin Xu et al, http://arxiv.org/pdf/1502.03044v2.pdf
8. RNN Tutorial Nervana Systems, https://www.nervanasys.com/recurrent-neural-networks/
9. Cho, NMT Tutorial http://nlp.stanford.edu/projects/nmt/Luong-Cho-Manning-NMT-ACL2016-v4.pdf
10. Word2Vec Tutorial https://www.tensorflow.org/tutorials/word2vec