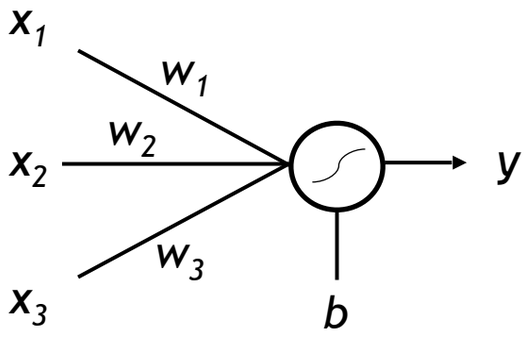
Recurrent Neural Networks

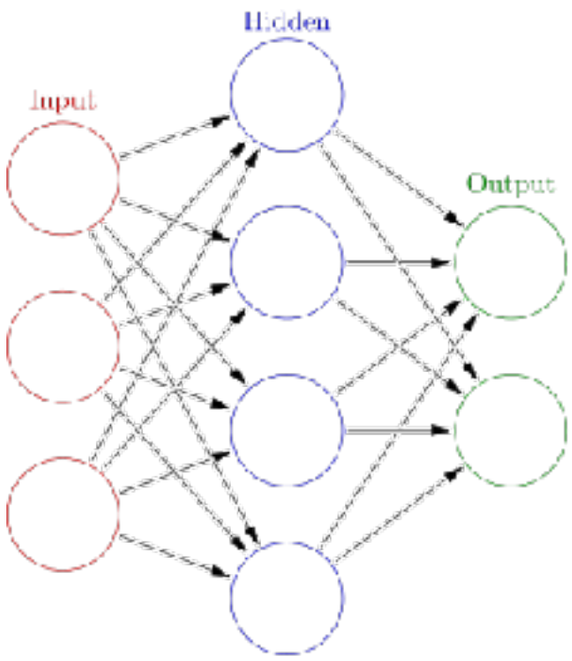
Pramod Kompalli
OLA

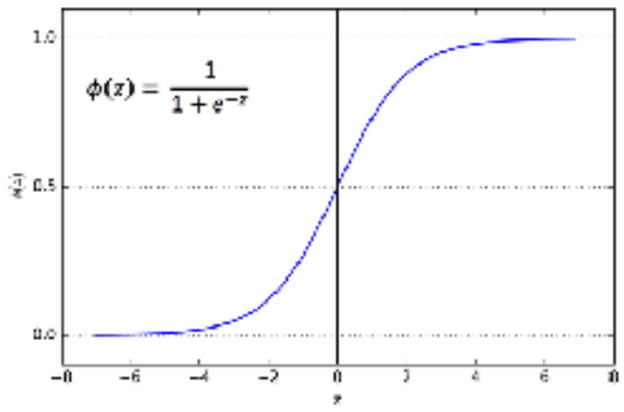
Recap

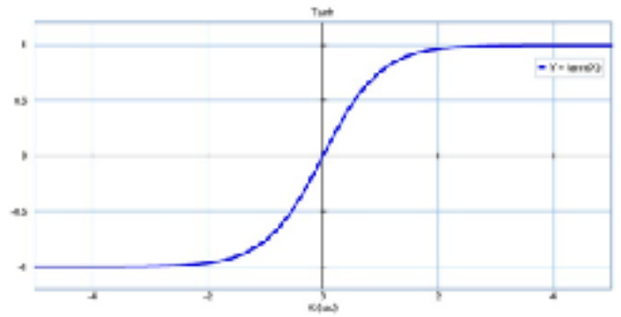


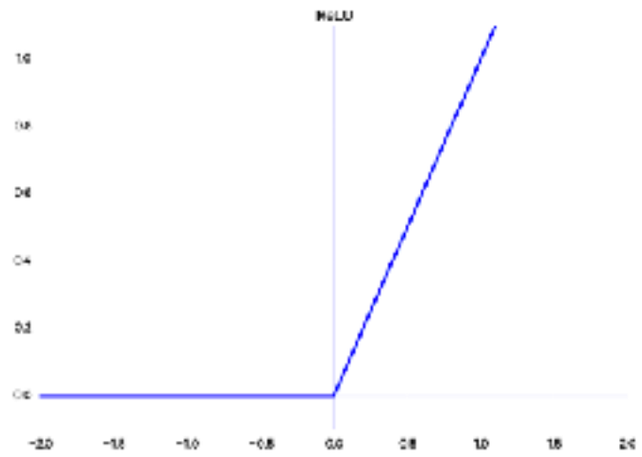
$$y = \begin{cases} 1, & g(w.x + b) > 0 \\ 0, & otherwise \end{cases}$$

$g(\cdot)$ *is a non-linear function*









Videos

YouTube

hotstar

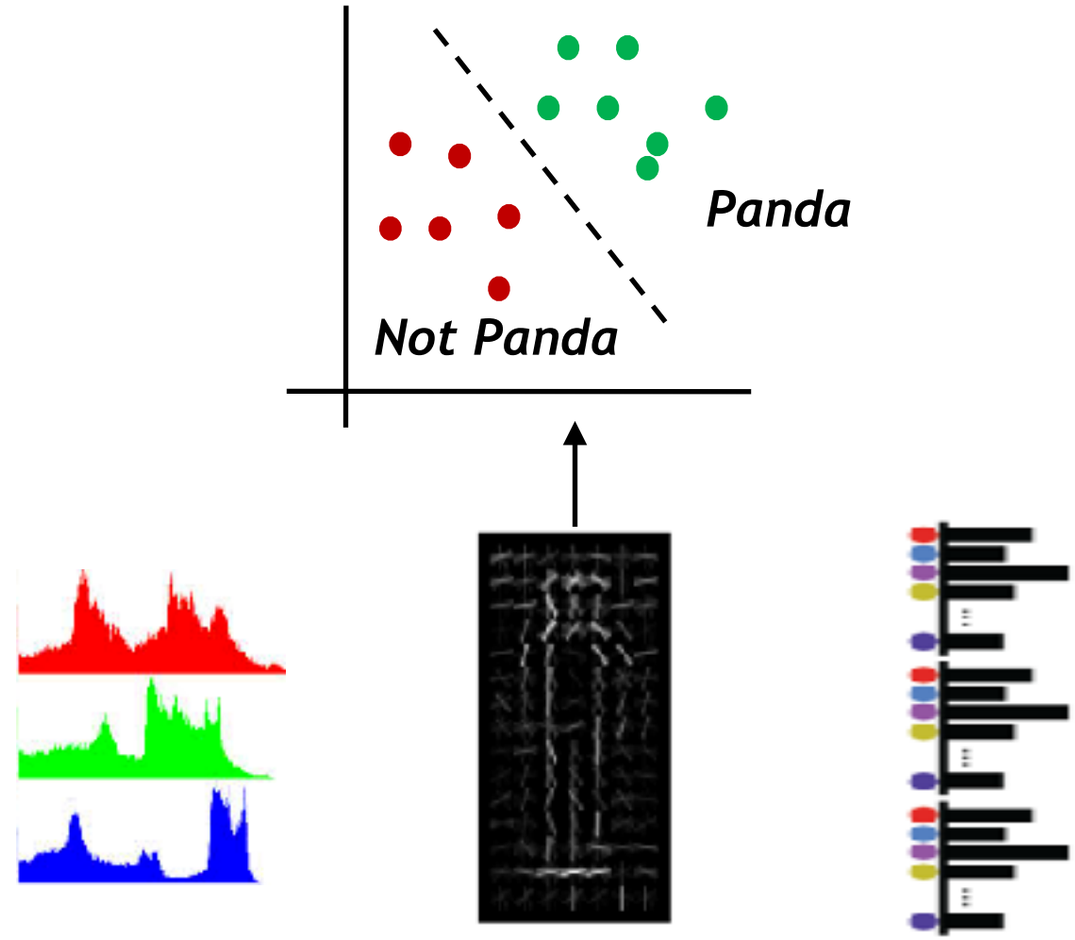
Videos


coursera
Decision Trees
Emily Fox & Carlos Guestrin
Machine Learning Specialization
University of Washington

Image vs. Video

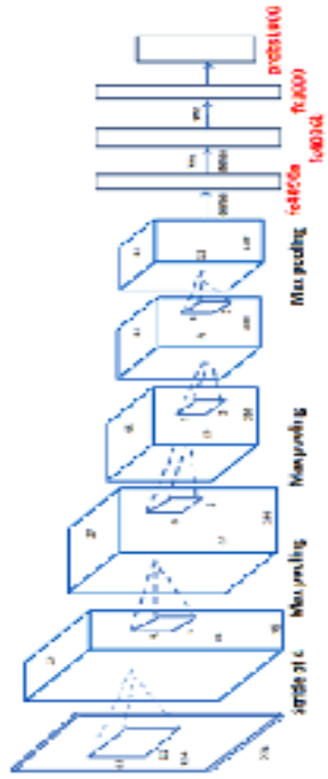
- Arbitrary length in T
- Scale
- Deeper Semantics

Representing Images

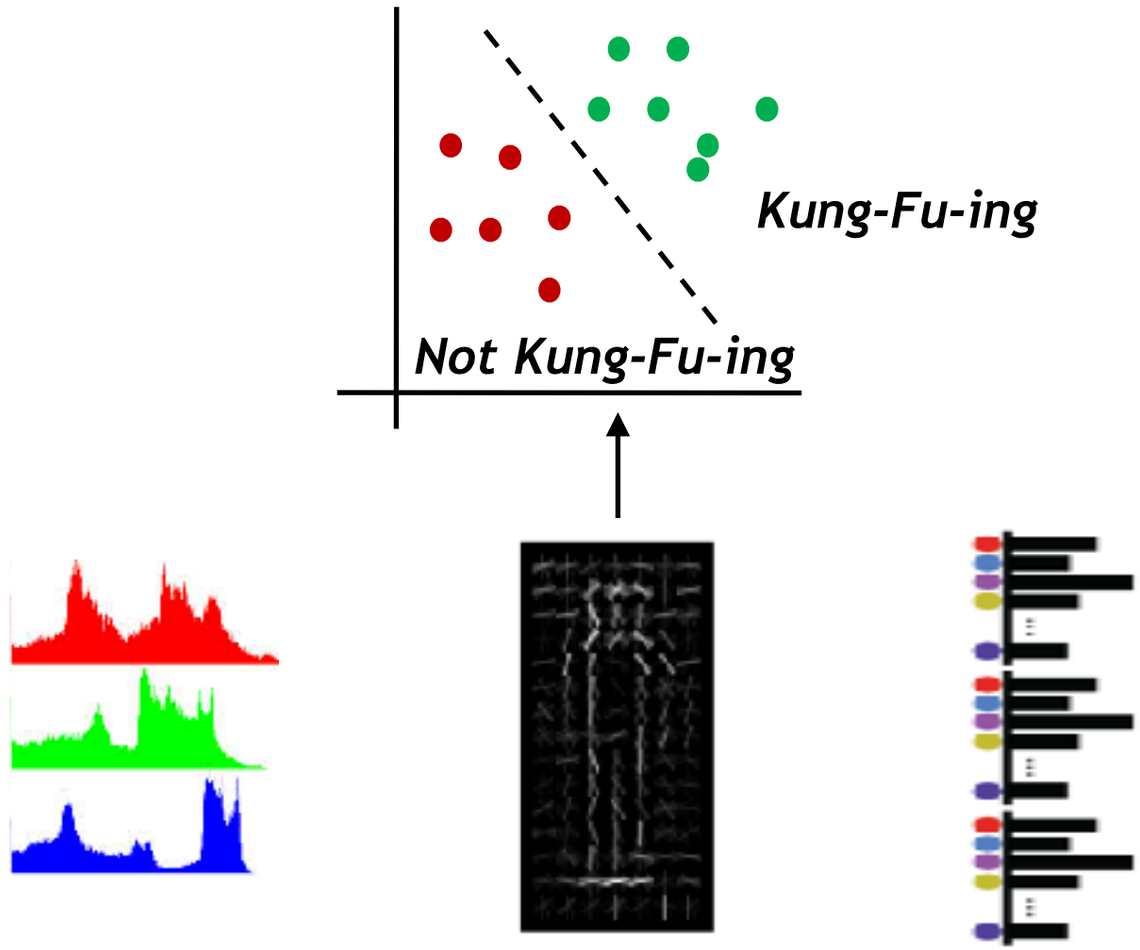


Modern Image Rep.

Panda

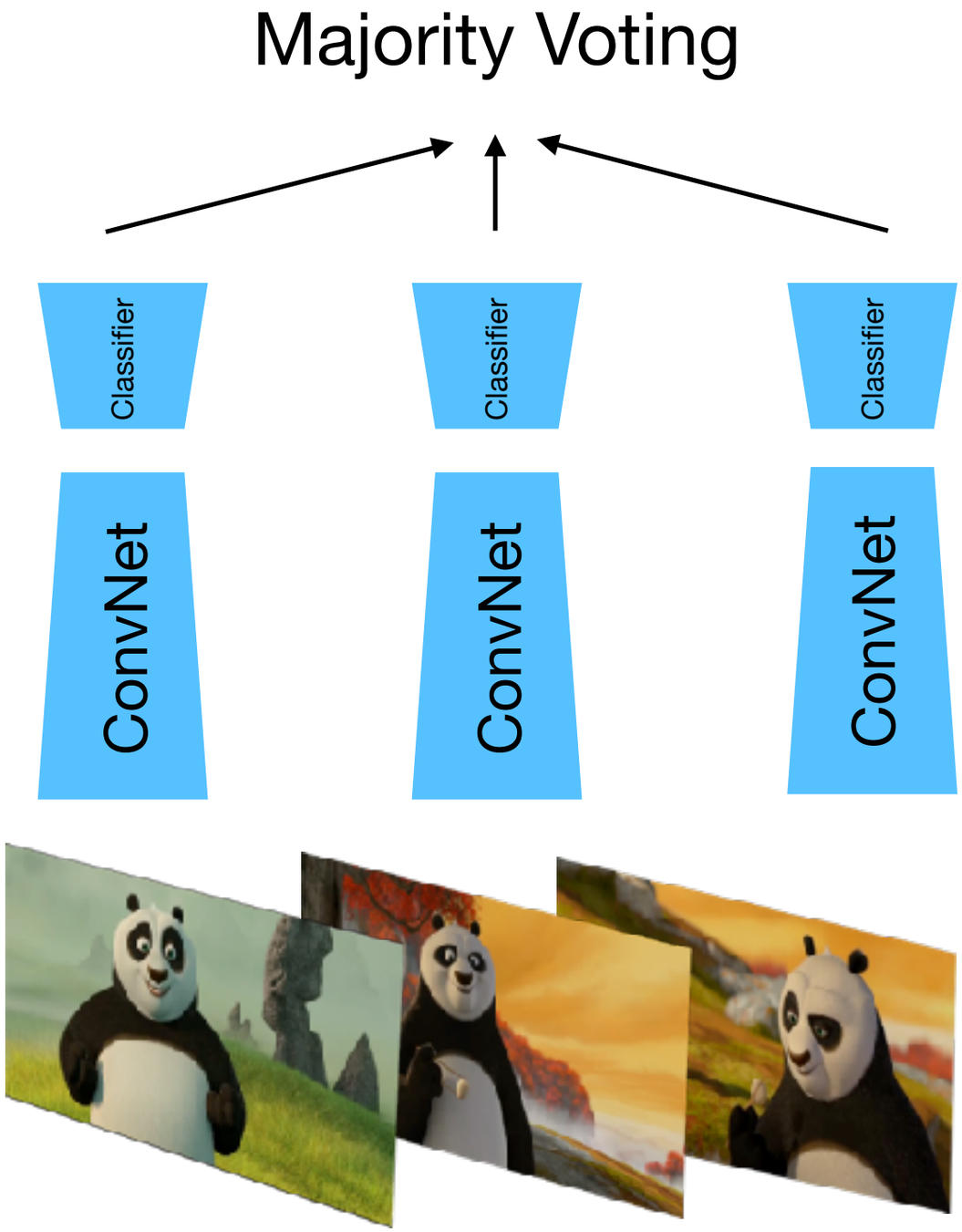


Simple Video Rep.

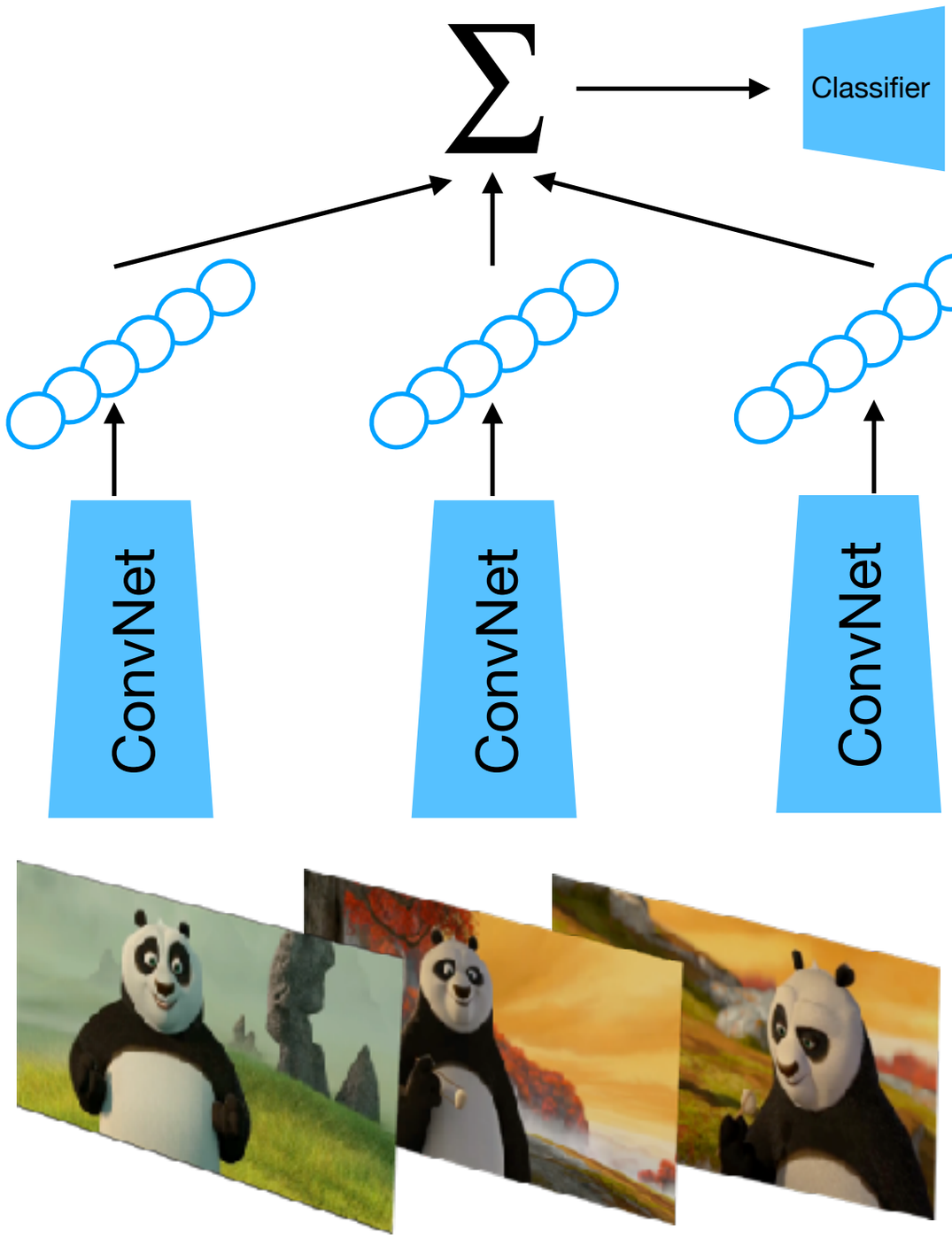


What could be the issues?

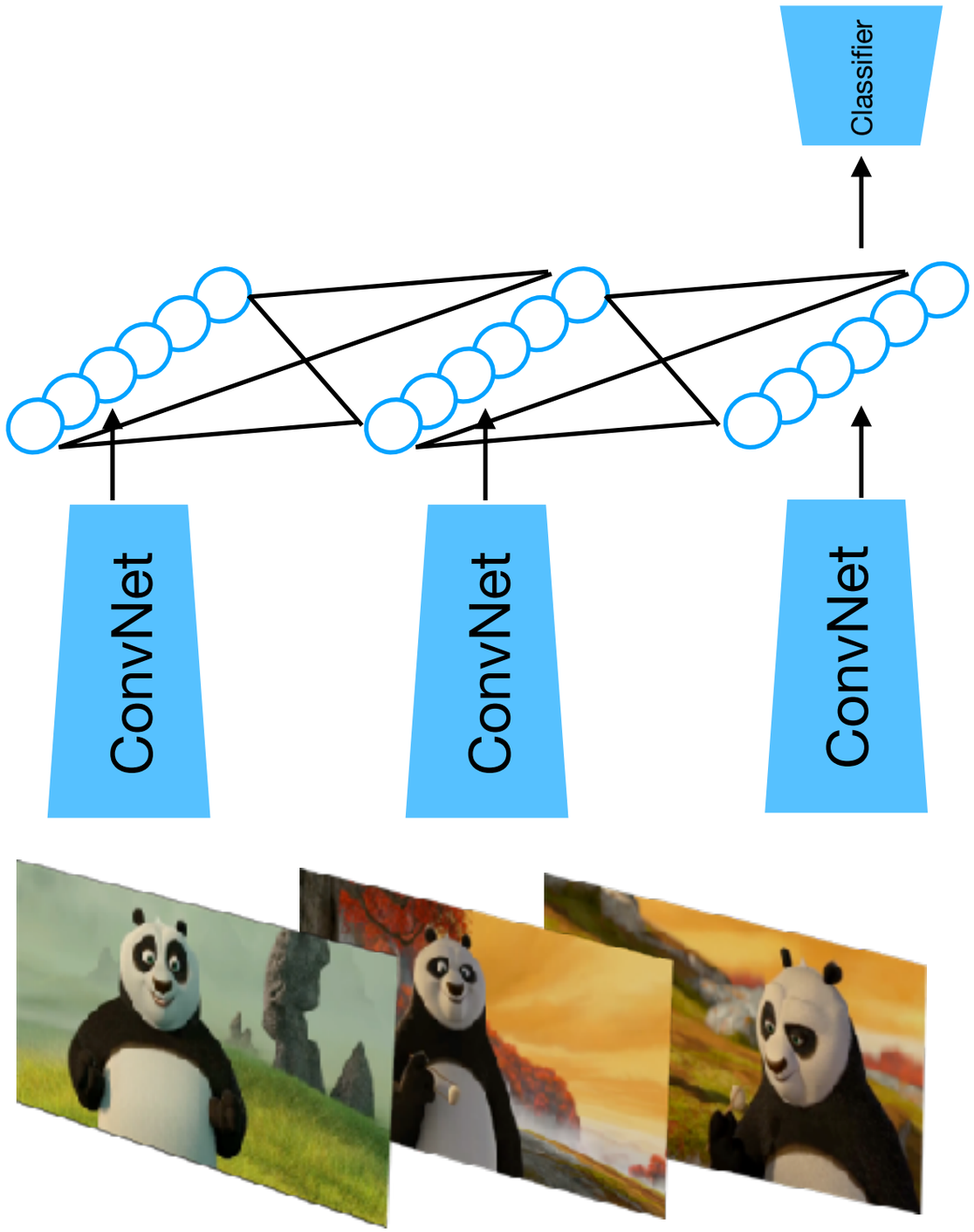
Late Fusion


Majority Voting
Classifier
Classifier
Classifier
ConvNet
ConvNet
ConvNet

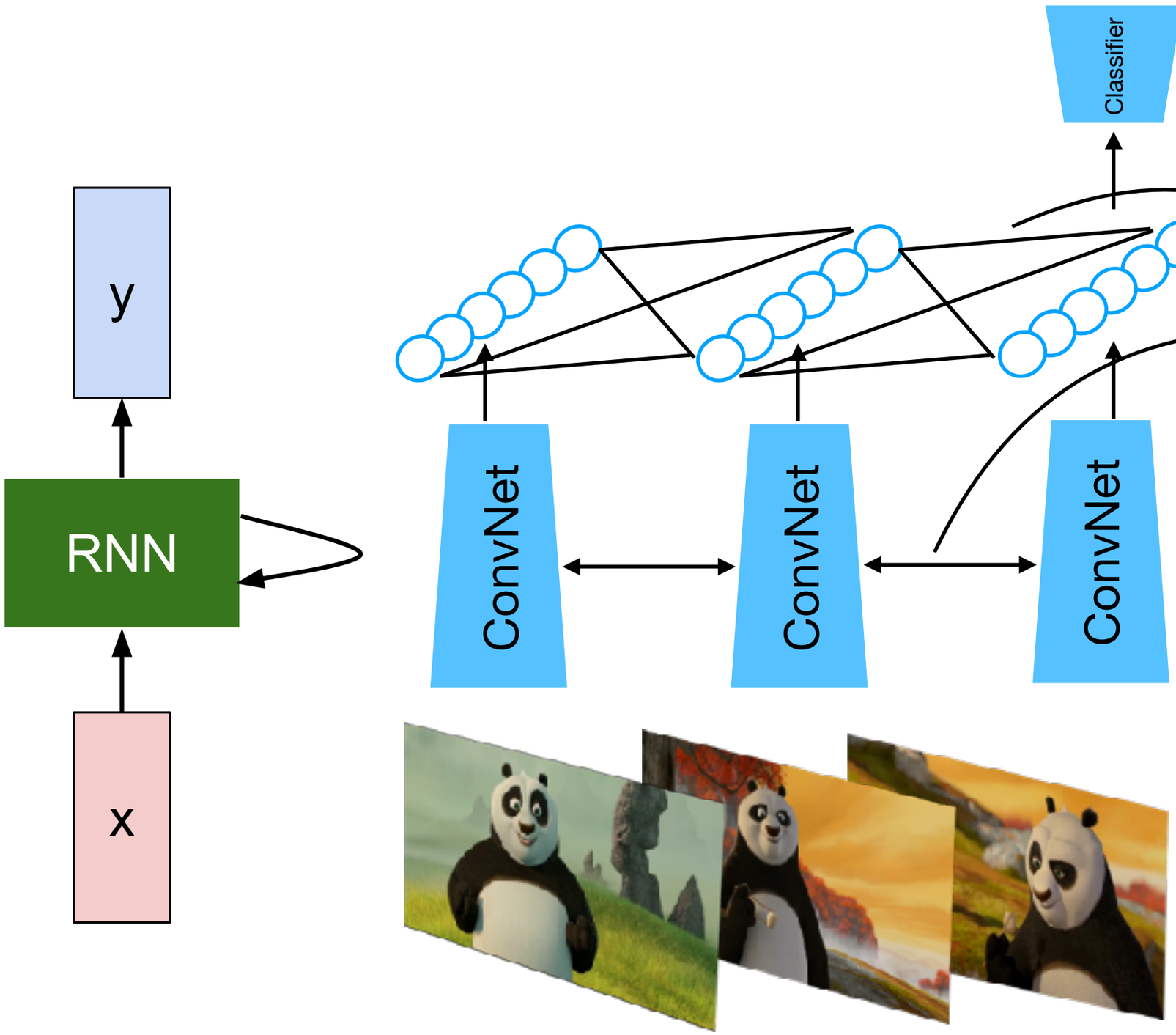
Slightly-Early Fusion



Fusing Representations

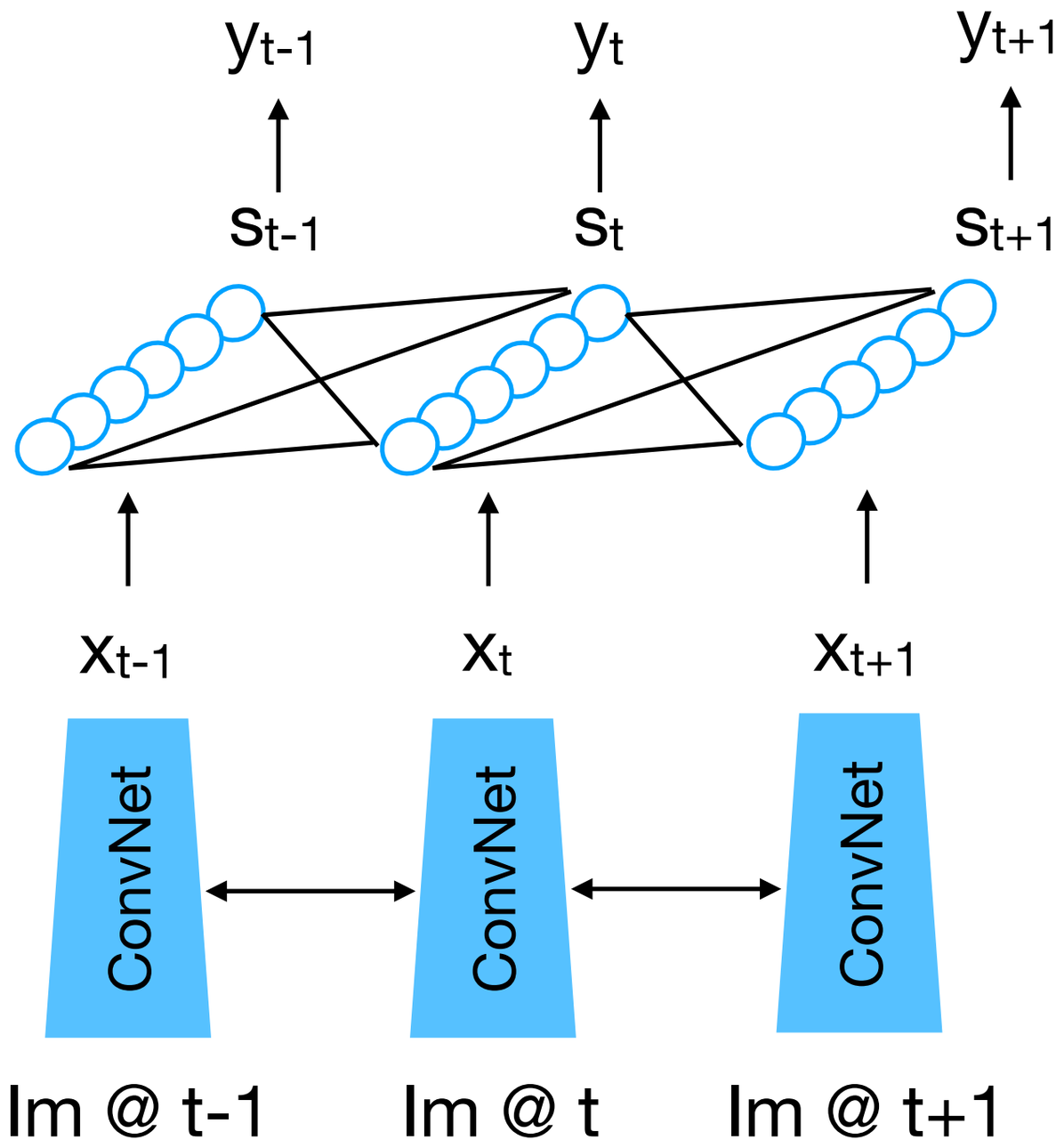


Fusing Representations

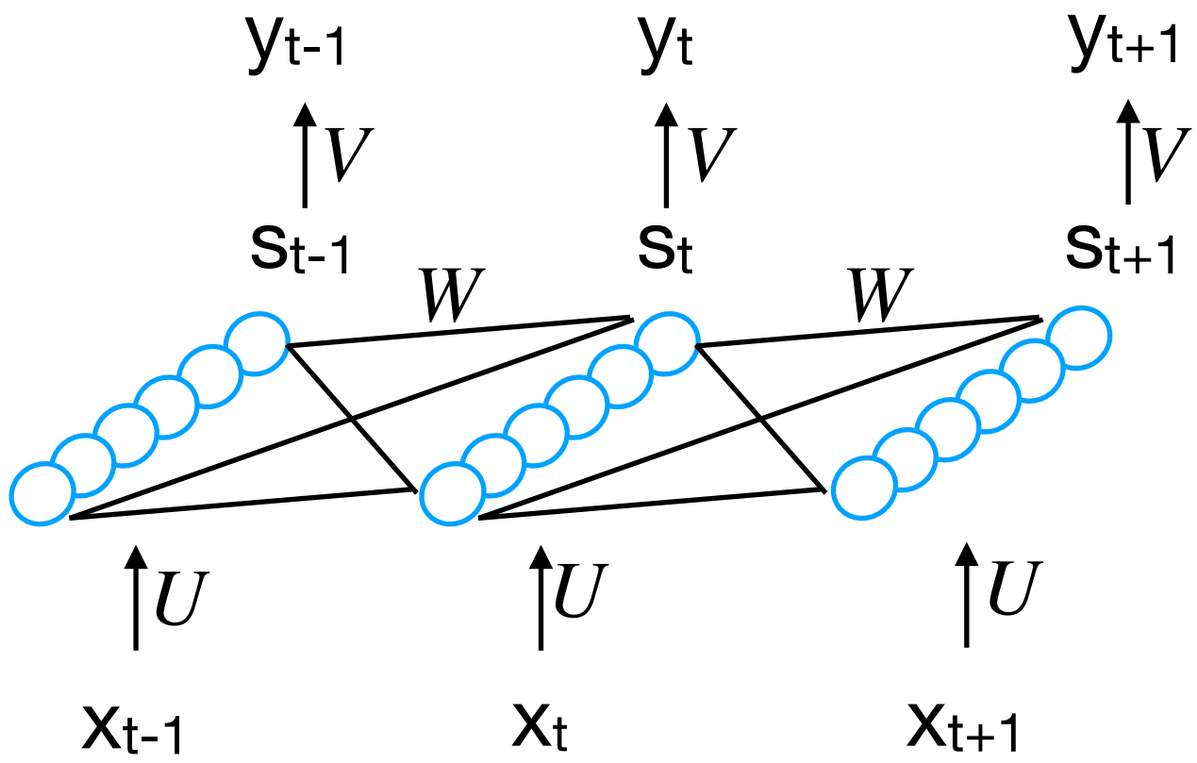


Unrolled View

RNN Formulation



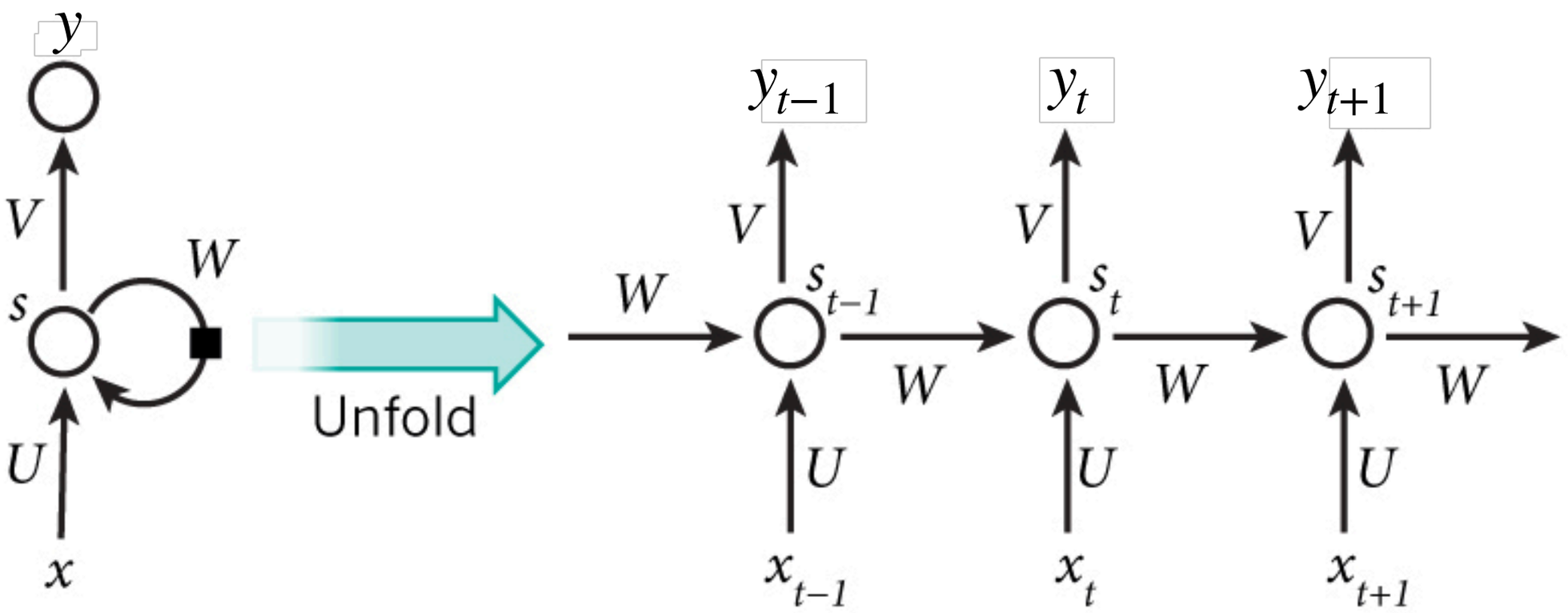
RNN Formulation



$$s_t = f_\theta(x_t, s_{t-1}) \qquad y_t = Vs_t$$

$$= tanh(Ux_t + Ws_{t-1})$$

RNN Formulation



$$s_t = f_W(x_t, s_{t-1})$$

$$= tanh(Ux_t + Ws_{t-1})$$

$$y_t = Vs_t$$

Depth in Layers
vs. Depth in Time

Training RNNs

- Typical loss: Cross Entropy (Predicted Label, True Label)

$$E_t = -\hat{y}_t \cdot log(y_t)$$

- Back-propagation

$$\frac{\partial E_3}{\partial V} = \frac{\partial E_3}{\partial y_3}\frac{\partial y_3}{\partial V}$$

$$= (y_3 - \hat{y}_3) \otimes s_3$$

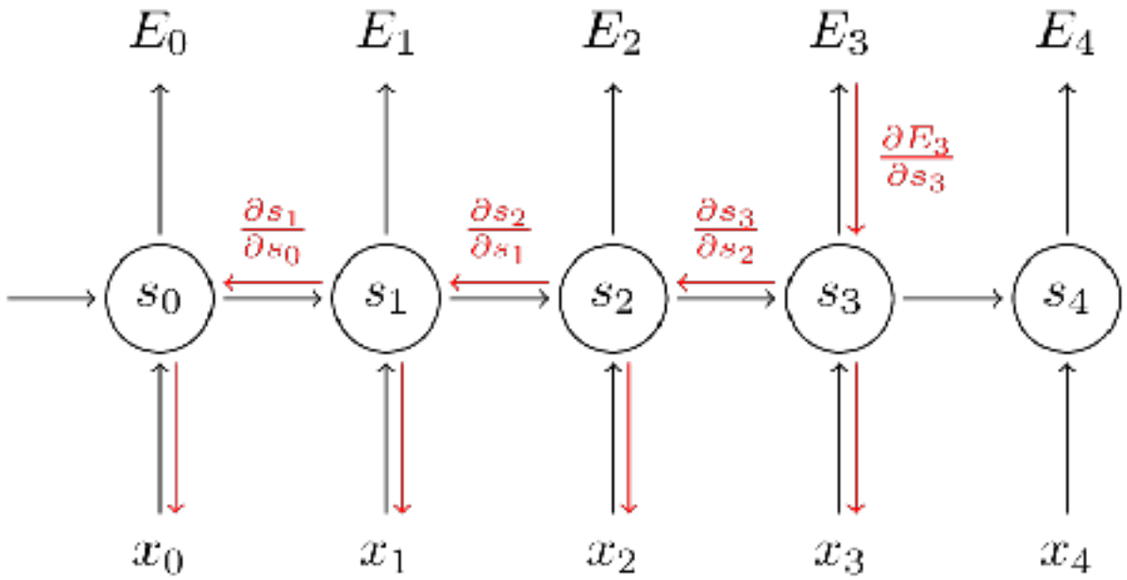
Training RNNs

- Back-propagation through Time (BPTT)

$$\frac{\partial E_3}{\partial W} = \frac{\partial E_3}{\partial y_3}\frac{\partial y_3}{\partial s_3}\frac{\partial s_3}{\partial W}$$

$$= \sum_{k=0}^{3} \frac{\partial E_3}{\partial y_3}\frac{\partial y_3}{\partial s_3}\frac{\partial s_3}{\partial s_k}\frac{\partial s_k}{\partial W}$$

$$= \sum_{k=0}^{3} \frac{\partial E_3}{\partial y_3}\frac{\partial y_3}{\partial s_3}\left[\prod_{j=k+1}^{3} \frac{\partial s_j}{\partial s_{j-1}}\right]\frac{\partial s_k}{\partial W}$$



$$\frac{\partial s_3}{\partial s_1} = \frac{\partial s_3}{\partial s_2}\frac{\partial s_2}{\partial s_1}$$

Vanishing/Exploding Gradients

$$\prod_{j=k+1}^{3} \frac{\partial s_j}{\partial s_{j-1}}$$

If Scalar?

If Gradient $\rightarrow$ 0, no information flows back
long term information is lost

If Gradient $\rightarrow \infty$, information overflows
historical information is lost
(could be handled with “Clipping”)

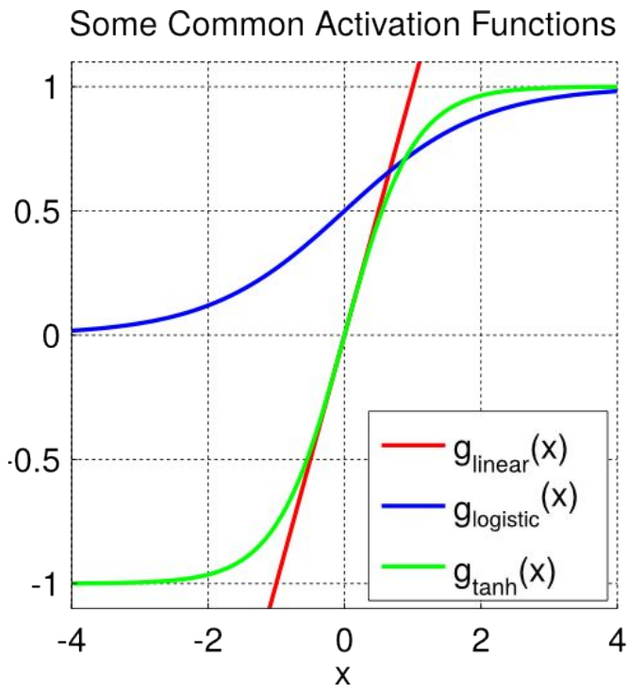
Vanishing/Exploding Gradients

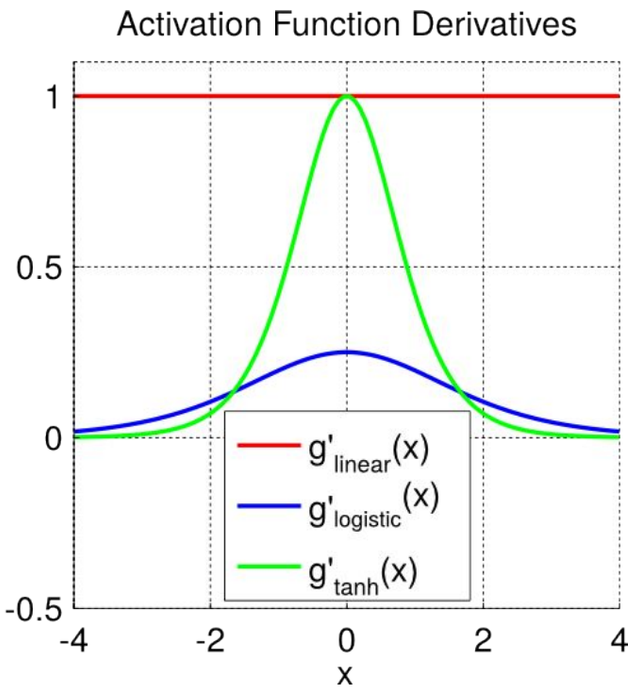
$$\frac{d\mathbf{f}}{d\mathbf{x}} = \left[\frac{\partial \mathbf{f}}{\partial x_1} \quad \cdots \quad \frac{\partial \mathbf{f}}{\partial x_n}\right] = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

$$\left\| \frac{\partial h_t}{\partial h_k} \right\| = \left\| \prod_{j=k+1}^{t} \frac{\partial h_j}{\partial h_{j-1}} \right\| \leq (\beta_W \beta_h)^{t-k}$$

Vanishing/Exploding Gradients

- Tanh is slightly better, derivative bounded by 1
- Sigmoid is bounded by 1/4
- ReLU?
  - Need good init
- Just use Better Units

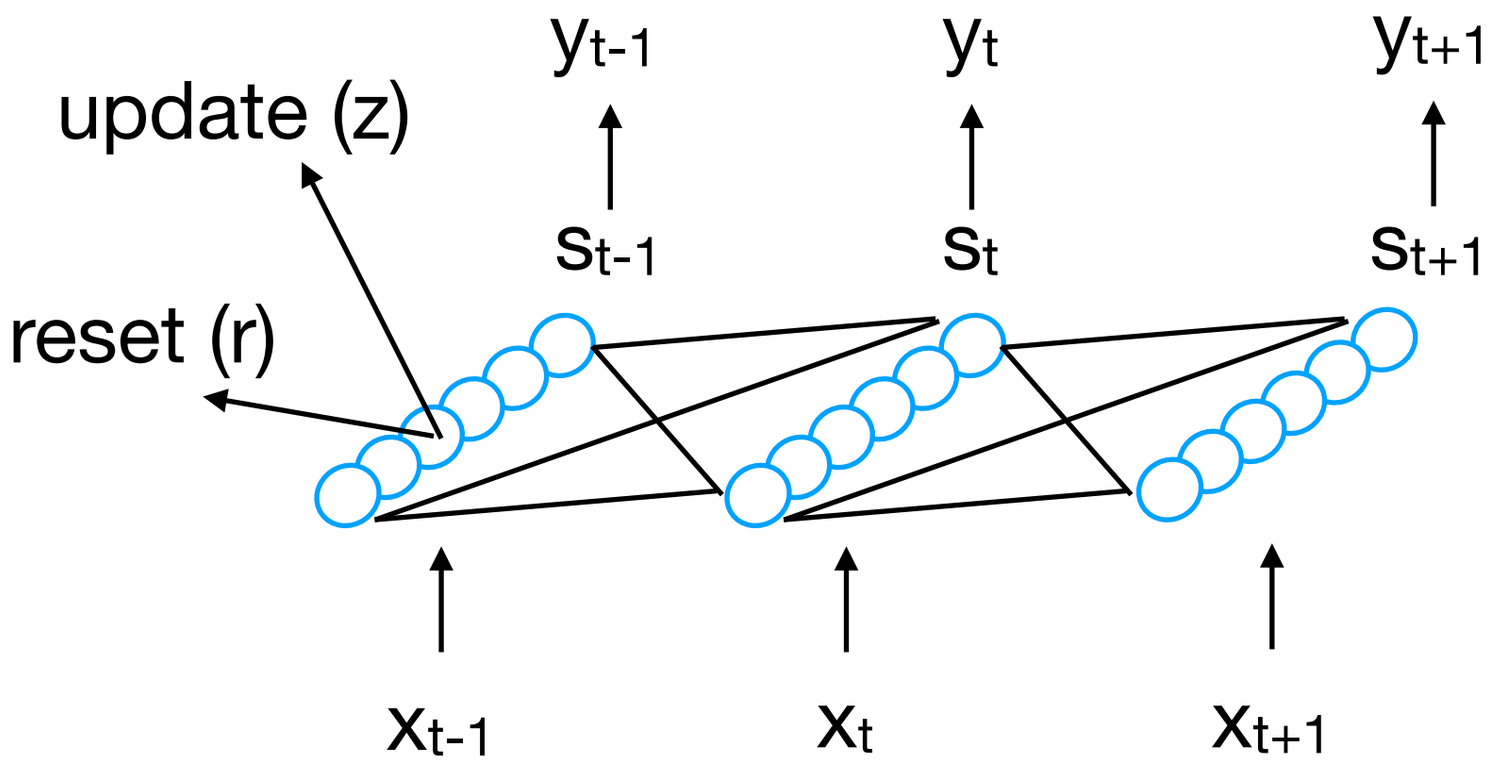




Picture courtesy: Girish Varma

Gated Recurrent Units

- Core Idea: Each “Unit” or neuron keeps track of a different amount of “History”



Gated Recurrent Units

- Core Idea: Each “Unit” or neuron keeps track of a different amount of “History”

- Update Gate: How much of previous state vector we want to keep

$$z = \sigma(x_t U^z + s_{t-1} W^z)$$

- Reset Gate: How to combine current input with previous state vector

$$r = \sigma(x_t U^r + s_{t-1} W^r)$$

Gated Recurrent Units

$$z = \sigma(x_t U^z + s_{t-1} W^z)$$

$$r = \sigma(x_t U^r + s_{t-1} W^r)$$

$$g = tanh(x_t U^g + (s_{t-1} \circ r) W^g)$$

$$s_t = (1 - z) \circ g + z \circ s_{t-1}$$

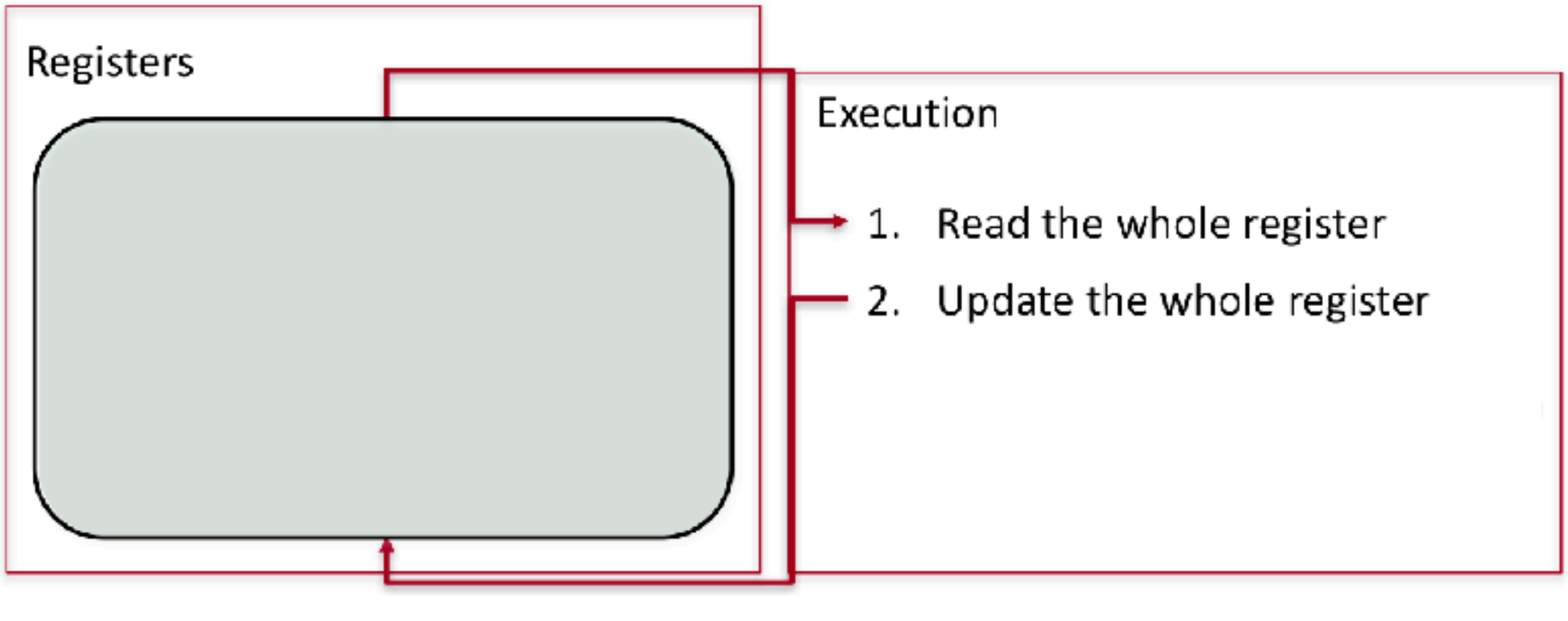
$$s_t = tanh(x_t U + s_{t-1} W)$$ Vanilla RNN Unit

Gated Recurrent Units

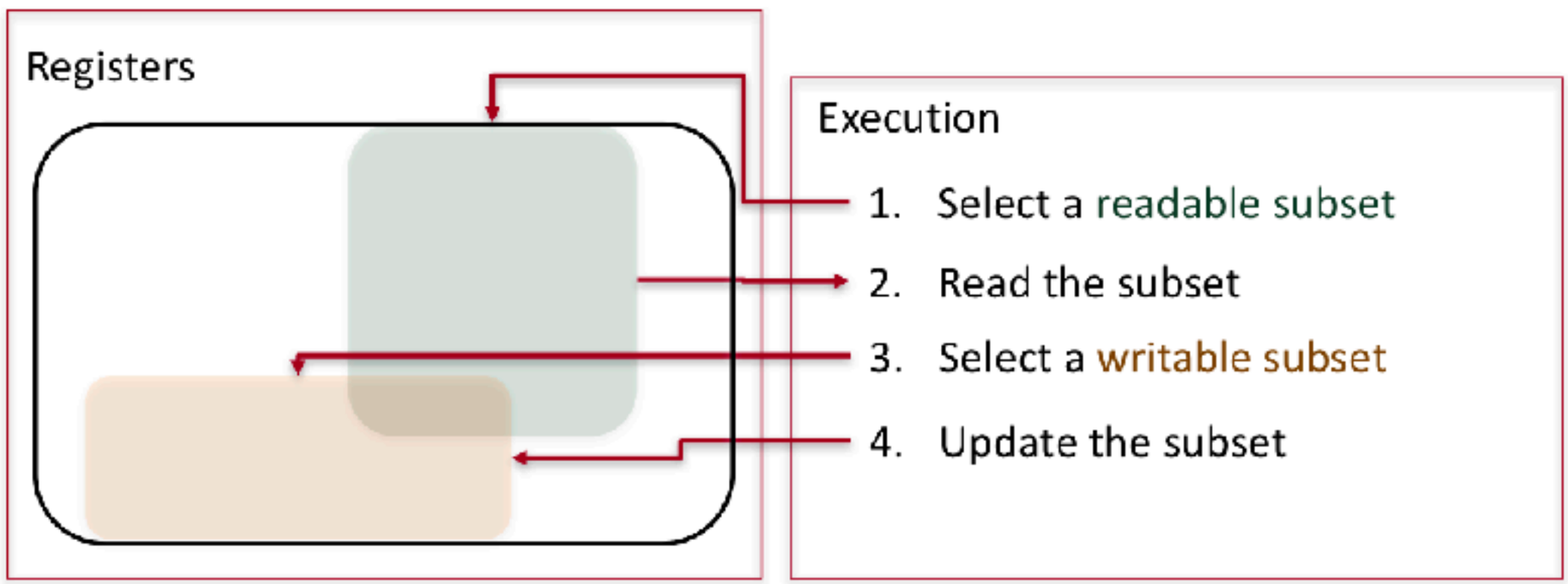
If Reset Gate → 0, ignore previous state,
use only current input
History is forgotten

If Update Gate → 1, ignore current input,
simply copy previous state
Gradient doesn't vanish

Vanilla RNN



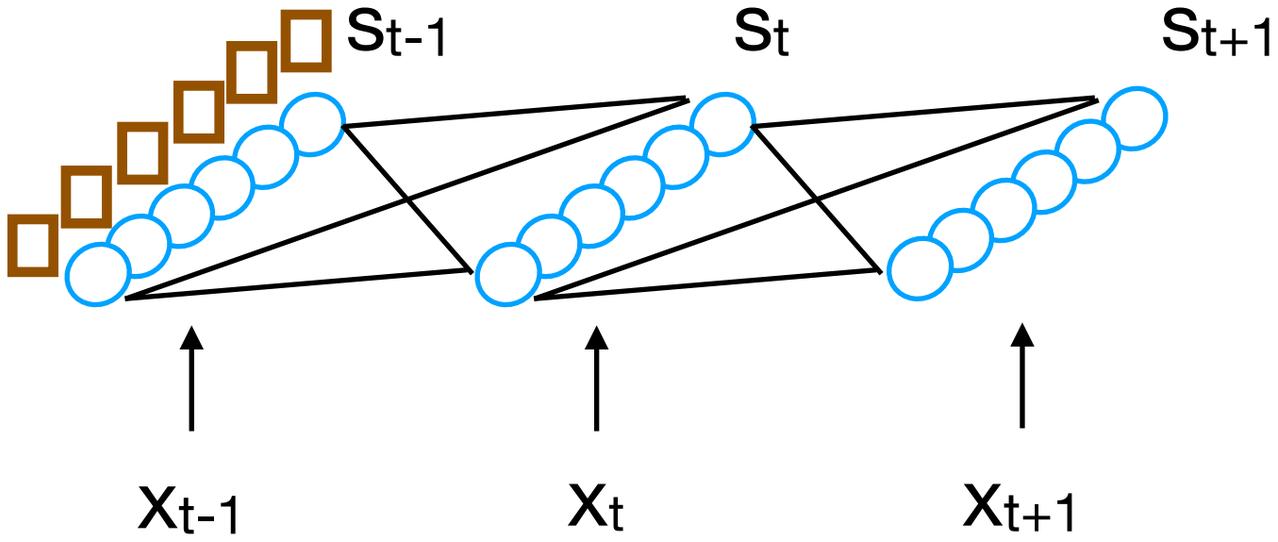
GRU



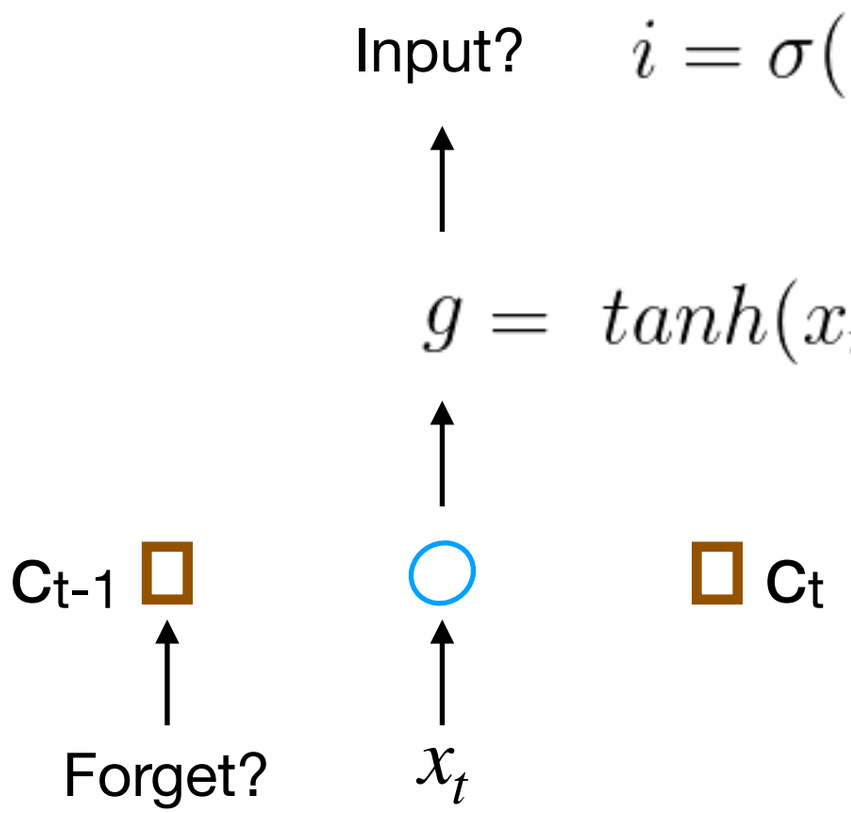
Picture courtesy: Richard Socher

Long-Short Term Memory

- Core Idea: Each “Unit” or neuron has a “Memory” cell in it

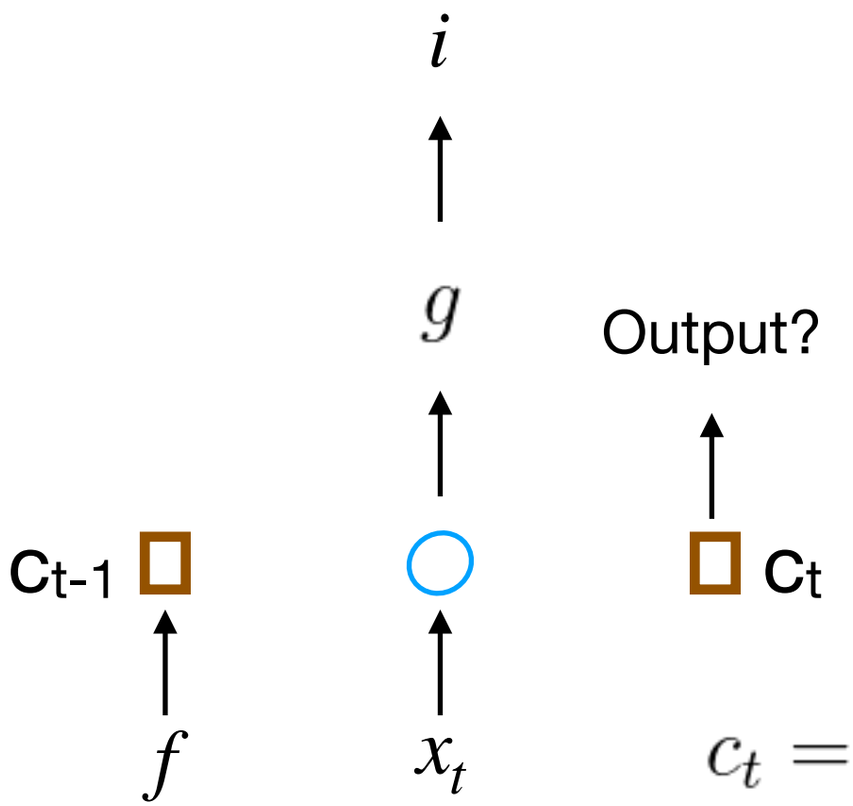


LSTM Unit



$$f = \sigma(x_t U^f + s_{t-1} W^f)$$

LSTM Unit



$$o = \sigma(x_t U^o + s_{t-1} W^o)$$

$$c_t = c_{t-1} \circ f + g \circ i$$

$$s_t = \tanh(c_t) \circ o$$

LSTM Unit

- Input Gate: How much of current input to use
- Forget Gate: How much of previous memory to use
- Output Gate: How much of current memory to use

- These gates explicitly model long-term dependencies
- Additive interaction

$$c_t = c_{t-1} \circ f + g \circ i$$

LSTM Unit

$$s_t = tanh(x_t U + s_{t-1} W)$$

Vanilla RNN Unit

$$g = \ tanh(x_t U^g + s_{t-1} W^g)$$

$$i = \sigma(x_t U^i + s_{t-1} W^i)$$

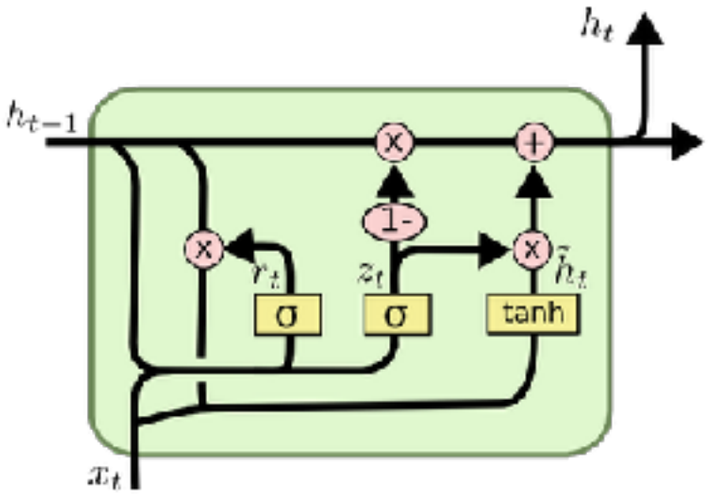
$$f = \sigma(x_t U^f + s_{t-1} W^f)$$

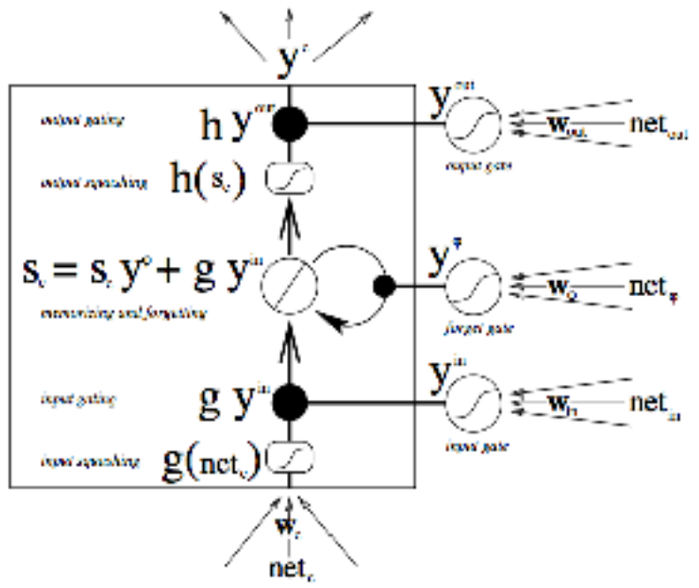
$$o = \sigma(x_t U^o + s_{t-1} W^o)$$

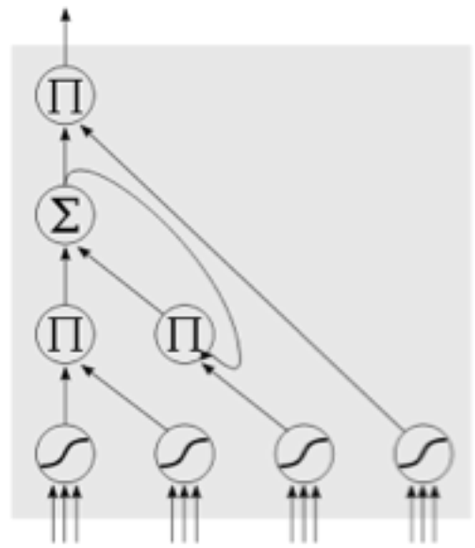
$$c_t = c_{t-1} \circ f + g \circ i$$

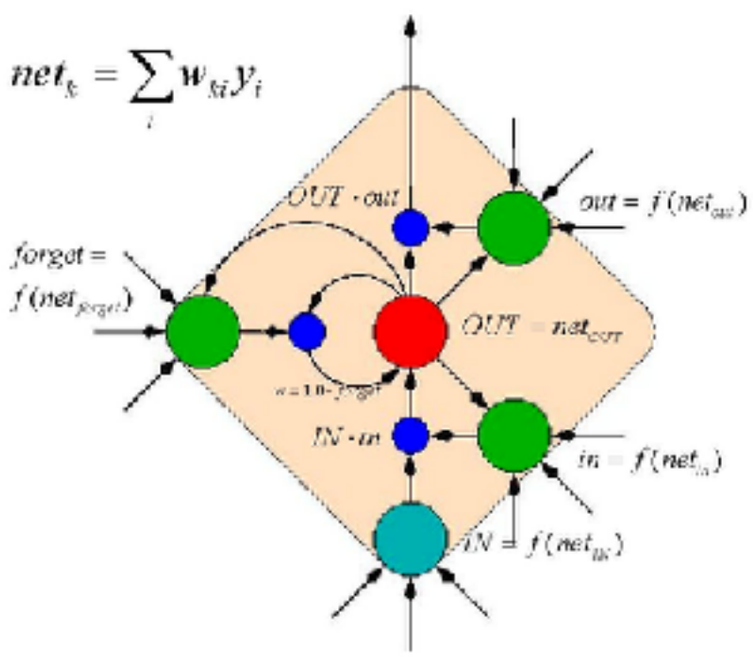
$$s_t = \tanh(c_t) \circ o$$

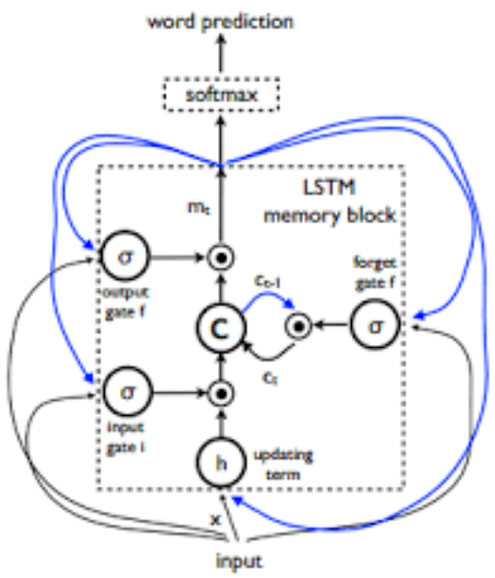
LSTM Unit

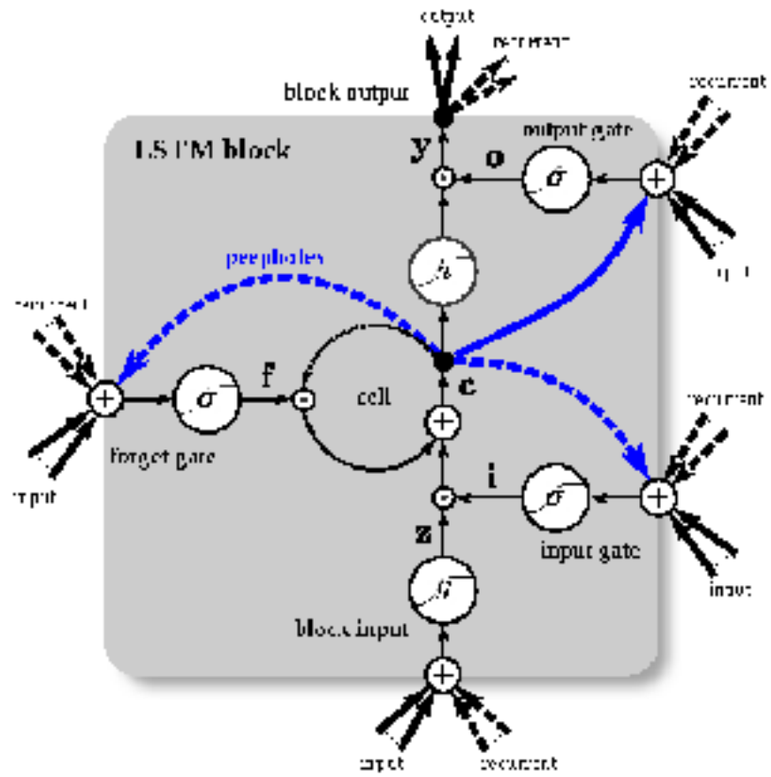

h_t
h_{t-1}
x_t
r_t
z_t
$\tilde{h}_t$
σ
σ
tanh
1-


y^c
h y^{in}
h(s_c)
$s_c = s_c\, y^\phi + g\, y^{in}$
g y^{in}
g(net_c)
w_c
net_c
y^{out}
w_{out}
net_{out}
y^ϕ
w_ϕ
net_ϕ
y^{in}
w_{in}
net_{in}

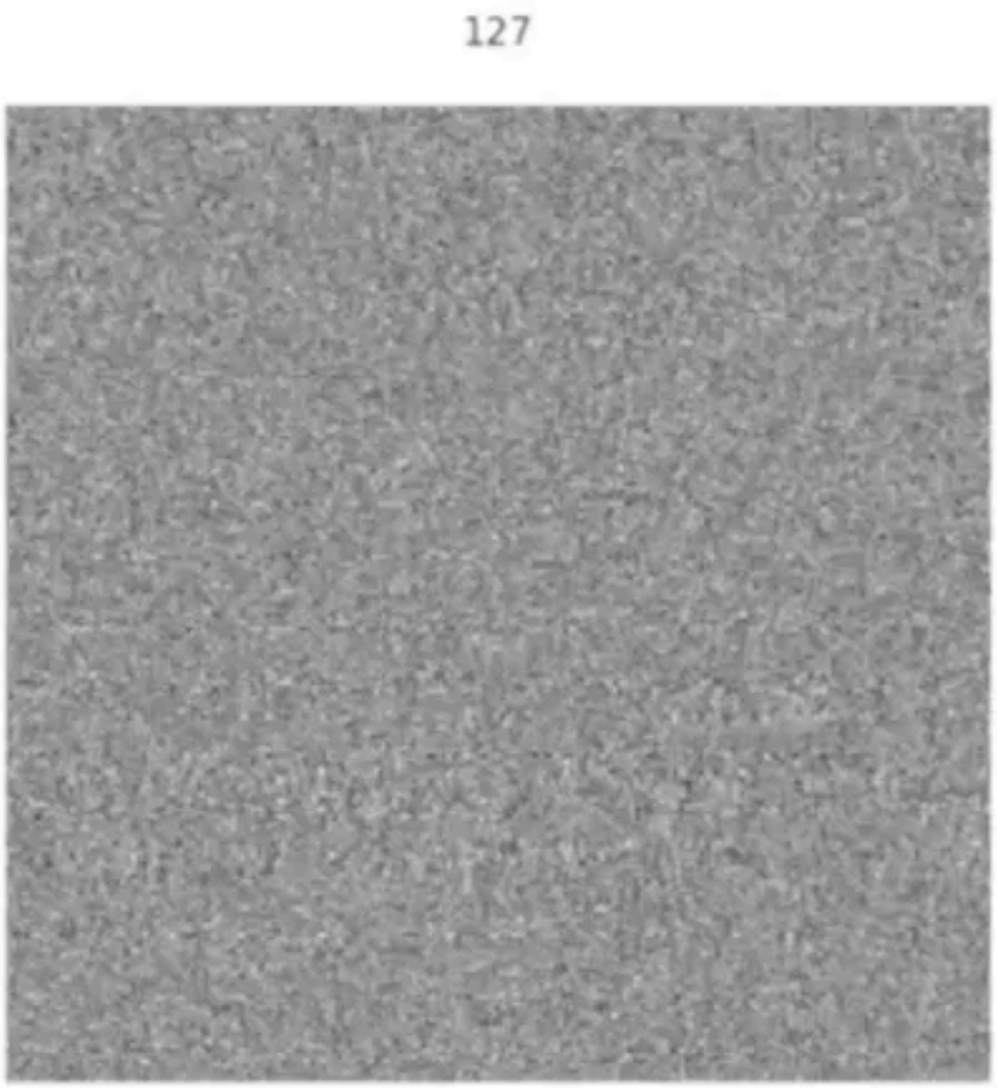

Π
Σ
Π
Π

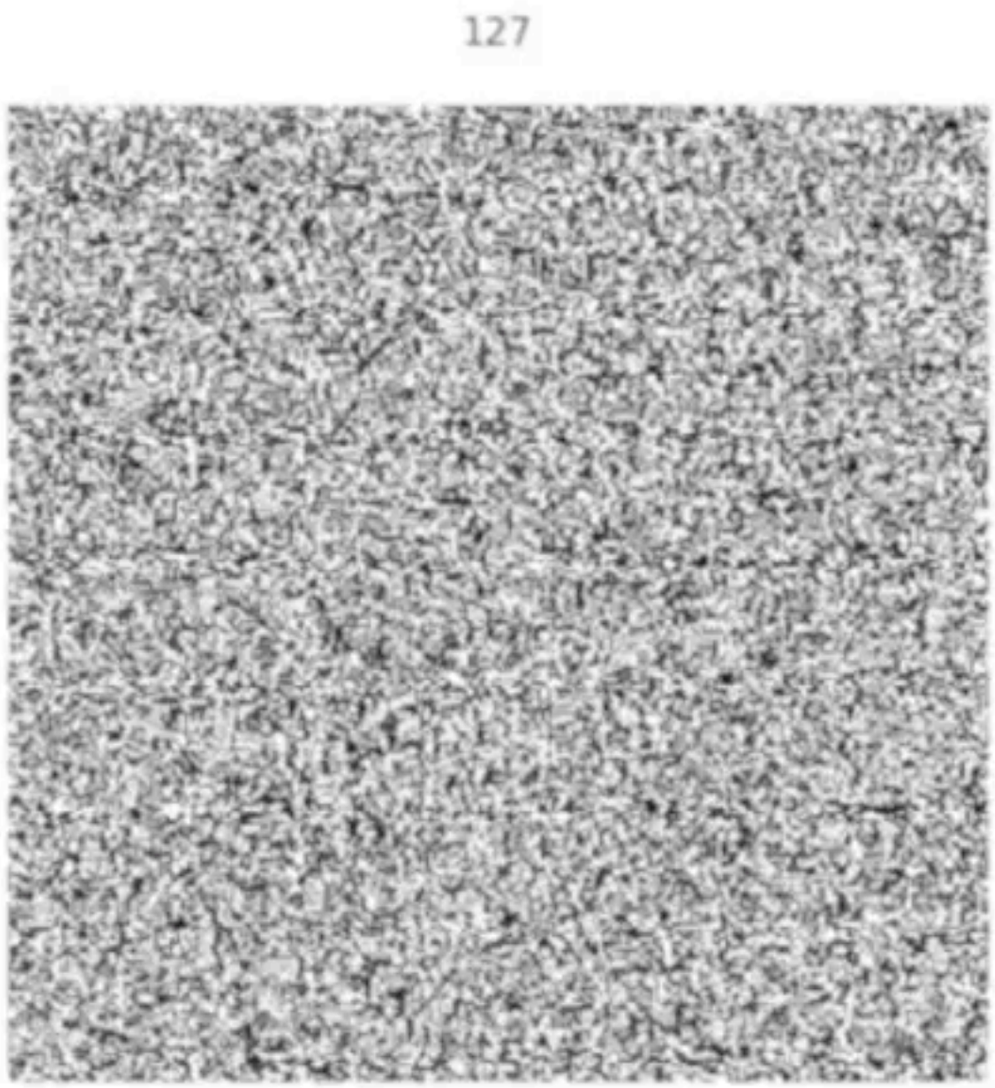

$net_k = \sum_i w_{ki} y_i$
OUT · out
out = f(net_out)
forget = f(net_forget)
OUT = net_OUT
IN · in
in = f(net_in)
IN = f(net_IN)


word prediction
softmax
LSTM memory block
output gate f
input gate i
updating term
forget gate f
input


output
block output
LSTM block
output gate
peepholes
forget gate
cell
input gate
block input
input

LSTM vs. RNN





https://imgur.com/gallery/vaNahKE

LSTM vs GRU

- 3 Gates
- Contains additional memory for each cell
- Two Non-Linearities in computing output
- More expressive power
  - Better results with more data

- 2 Gates
- No memory beyond cell state
- No second Non-Linearity
- Slightly lesser parameters
  - Quicker to Train
  - Smaller datasets

LSTM Training Advice

- Use an LSTM or GRU: it makes your life so much simpler!
- Initialize recurrent matrices to be orthogonal
- Initialize other matrices with a sensible (small!) scale
- Initialize forget gate bias to 1: default to remembering
- Use adaptive learning rate algorithms: Adam, AdaDelta, ...
- Clip the norm of the gradient: 1–5 seems to be a reasonable threshold when used together with Adam or AdaDelta.
- Either only dropout vertically or learn how to do it right
- Be patient!

RNN Architectures

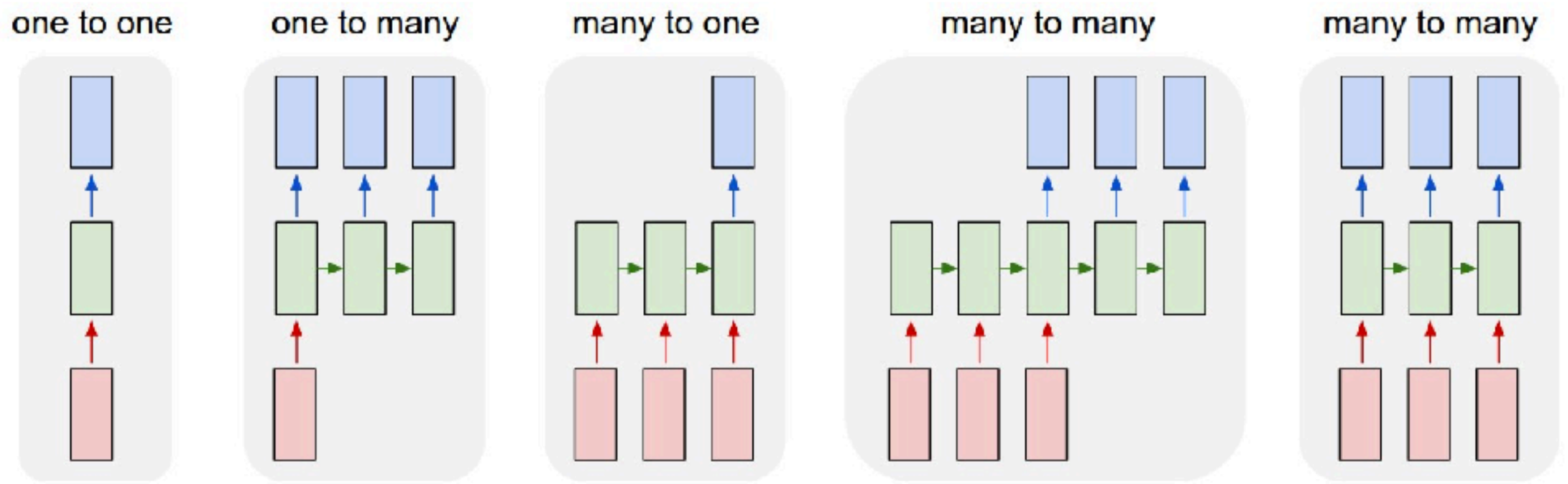


Image Classification

Object Recognition

Image Captioning

Action/Activity Recognition

Video Sentiment Analysis

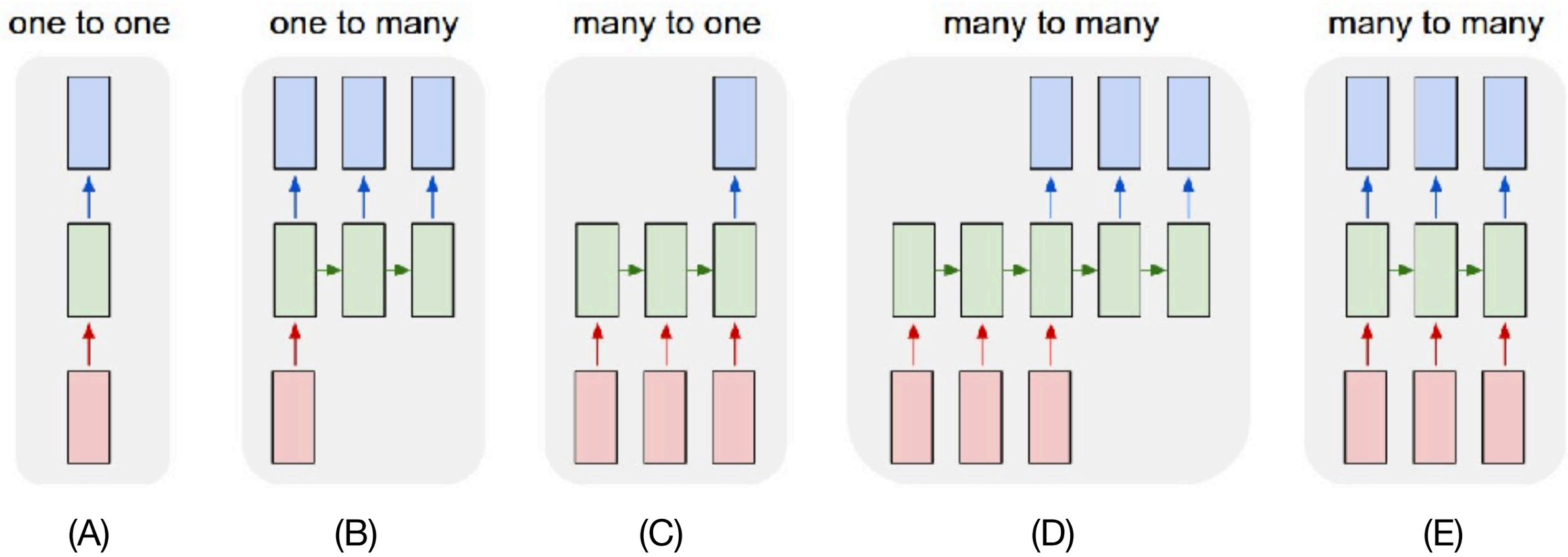
Video Captioning

Frame-level Labelling

Object Detection In Video

Picture courtesy: Andrej Karpathy

Beyond Vision



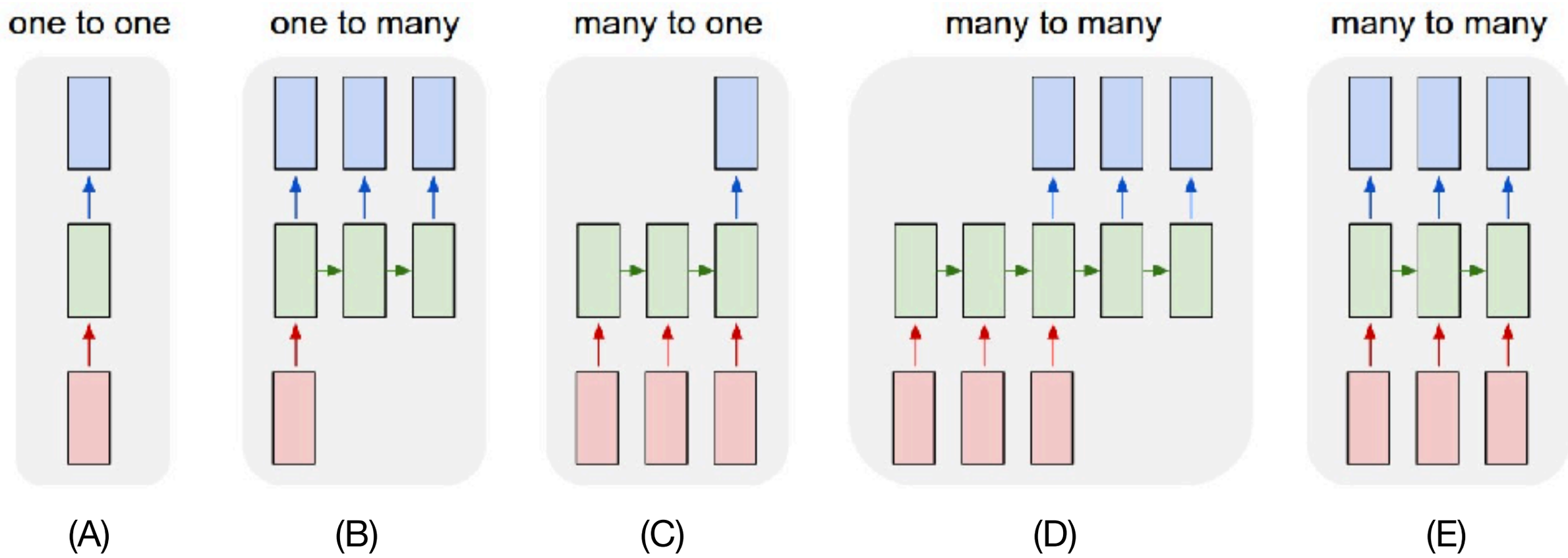
Speech Recognition

Speaker Recognition

Sentiment from a Tweet

Translate sentence from English to Telugu

Beyond Vision



Text to Speech (Alexa)

Answering a Question

Controlling a Self-Driving Car

Generate Trailer for a Movie

Modeling Text

- Given a partial sentence, predict the next word

My name is _____

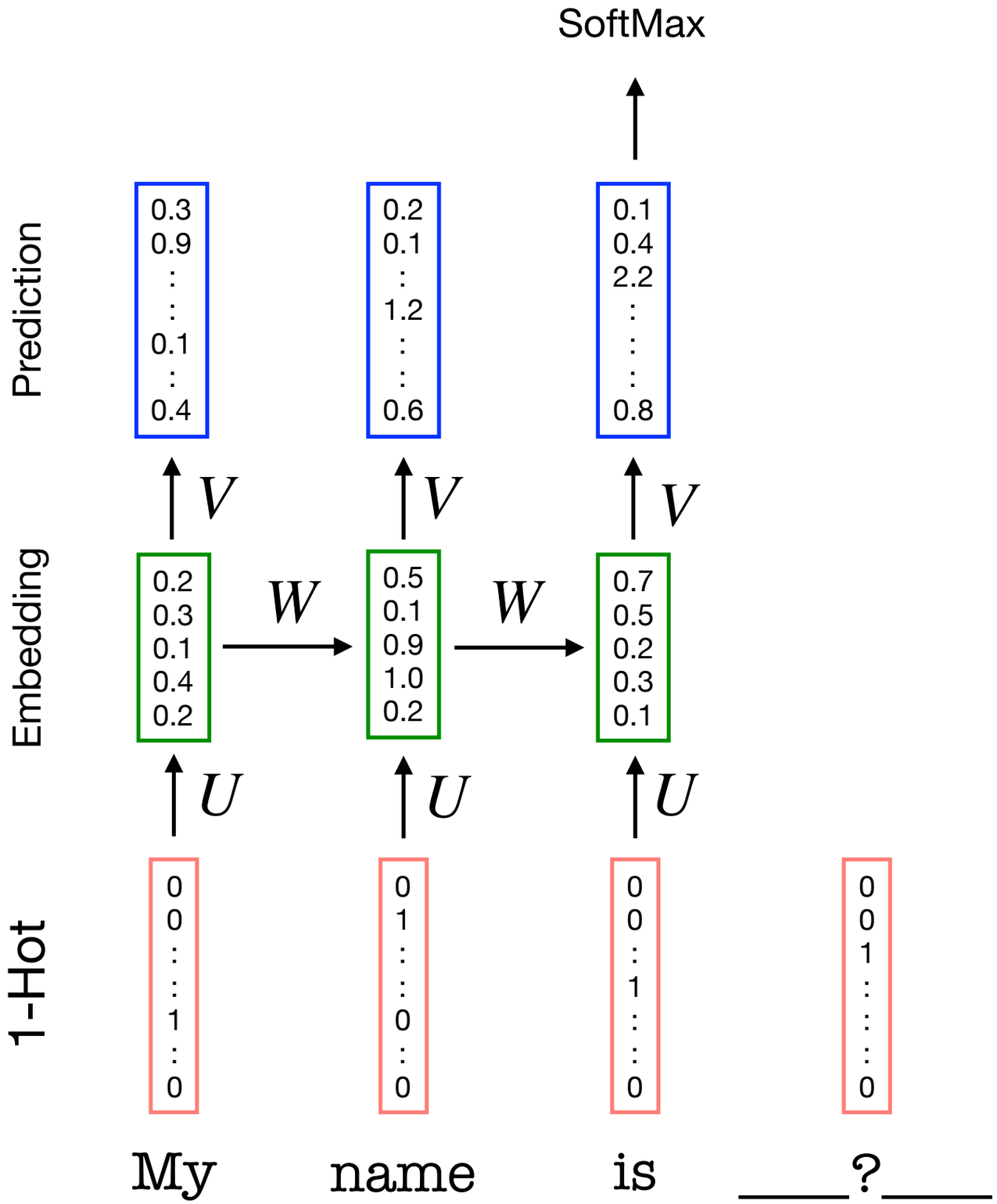
My name is Anthony _____

My name is Anthony Gonzalves _____

My name is Anthony Gonzalves main _____

My name is Anthony Gonzalves main duniya _____

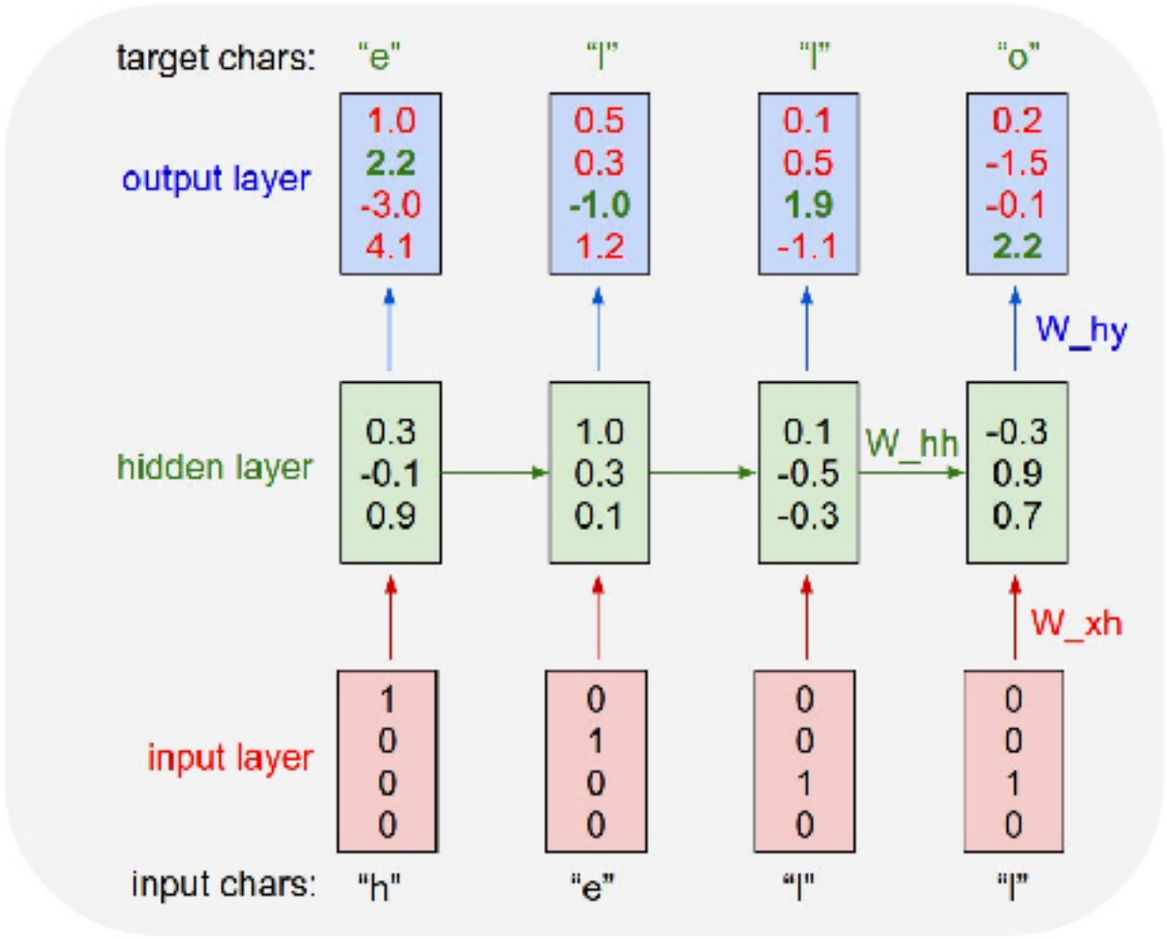
My name is Anthony Gonzalves main duniya mein _____


SoftMax
Prediction
0.3
0.9
0.1
0.4
0.2
0.1
1.2
0.6
0.1
0.4
2.2
0.8
V
V
V
Embedding
0.2
0.3
0.1
0.4
0.2
W
0.5
0.1
0.9
1.0
0.2
W
0.7
0.5
0.2
0.3
0.1
U
U
U
1-Hot
0
0
1
0
0
1
0
0
0
1
0
0
My
name
is
____?____

Char-RNN

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

Easier at Character level?



all hail andrej karpathy

The Shakespeare Hallucination

at first:

```
tyntd-iafhatawiaoihrdemot  lytdws  e ,tfti, astai f ogoh eoase rrranbyne 'nhthnee e
plia tklrgd t o idoe ns,smtt   h ne etie h,hregtrs nigtike,aoaenns lng
```

train more

```
"Tmont thithey" fomesscerliund
Keushey. Thom here
sheulke, anmerenith ol sivh I lalterthend Bleipile shuwy fil on aseterlome
coaniogennc Phe lism thond hon at. MeiDimorotion in ther thize."
```

train more

```
Aftair fall unsuch that the hall for Prince Velzonski's that me of
her hearly, and behs to so arwage fiving were to it beloge, pavu say falling misfort
how, and Gogition is so overelical and ofter.
```

train more

```
"Why do what that day," replied Natasha, and wishing to himself the fact the
princess, Princess Mary was easier, fed in had oftened him.
Pierre aking his soul came to the packs and drove up his father-in-law women.
```

all hail andrej karpathy

The Shakespeare Hallucination

```
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.
```

all hail andrej karpathy

Generated Wikipedia Text

```
Naturalism and decision for the majority of Arab countries' capitalide was grounded
by the Irish language by [[John Clair]], [[An Imperial Japanese Revolt]], associated
with Guangzham's sovereignty. His generals were the powerful ruler of the Portugal
in the [[Protestant Immineners]], which could be said to be directly in Cantonese
Communication, which followed a ceremony and set inspired prison, training. The
emperor travelled back to [[Antioch, Perth, October 25|21]] to note, the Kingdom
of Costa Rica, unsuccessful fashioned the [[Thrales]], [[Cynth's Dajoard]], known
in western [[Scotland]], near Italy to the conquest of India with the conflict.
Copyright was the succession of independence in the slop of Syrian influence that
was a famous German movement based on a more popular servicious, non-doctrinal
and sexual power post. Many governments recognize the military housing of the
[[Civil Liberalization and Infantry Resolution 265 National Party in Hungary]],
that is sympathetic to be to the [[Punjab Resolution]]
(PJS)[http://www.humah.yahoo.com/guardian.
cfm/7754800786d17551963s89.htm Official economics Adjoint for the Nazism, Montgomery
was swear to advance to the resources for those Socialism's rule,
was starting to signing a major tripad of aid exile.]]
```

all hail andrej karpathy

Generated C 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;
    }
  }
  rw->name = "Getjbbregs";
  bprm_self_clearl(&iv->version);
  regs->new = blocks[(BPF_STATS << info->historidac)] | PFMR_CLOBATHINC_SECONDS << 12;
  return segtable;
}
```

all hail andrej karpathy

Generated Algebraic Geometry

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

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

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* $\mathfrak{q}' = 0$.

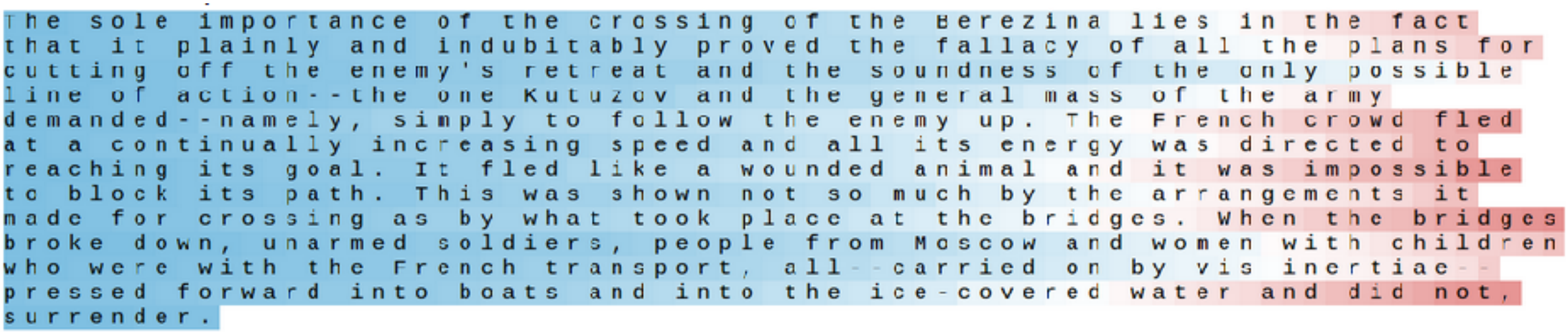
Proof. We will use the property we see that $\mathfrak{p}$ is the next functor (??). On the other hand, by Lemma ?? we see that

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

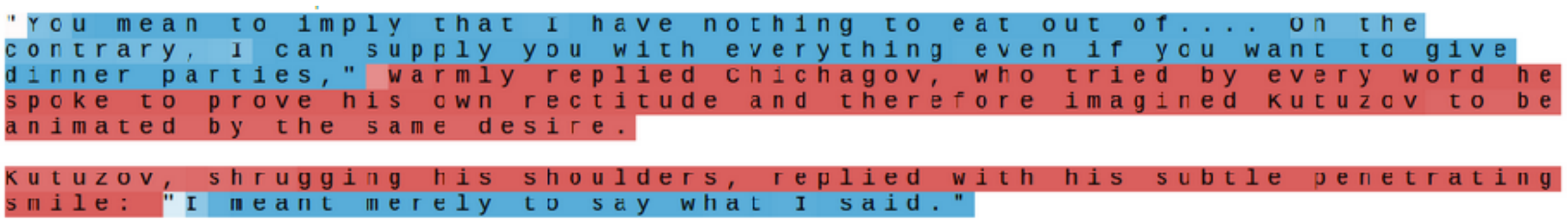
where K is an F-algebra where δ_{n+1} is a scheme over S. □

all hail andrej karpathy

Interpreting LSTM Cells



Length Tracker



Inside/Outside Quotes

all hail andrej karpathy

Interpreting LSTM Cells

```
static int __dequeue_signal(struct sigpending *pending, sigset_t *mask,
        siginfo_t *info)
{
  int sig = next_signal(pending, mask);
  if (sig) {
    if (current->notifier) {
      if (sigismember(current->notifier_mask, sig)) {
        if (!(current->notifier)(current->notifier_data)) {
          clear_thread_flag(TIF_SIGPENDING);
          return 0;
        }
      }
    }
    collect_signal(sig, pending, info);
  }
  return sig;
}
```

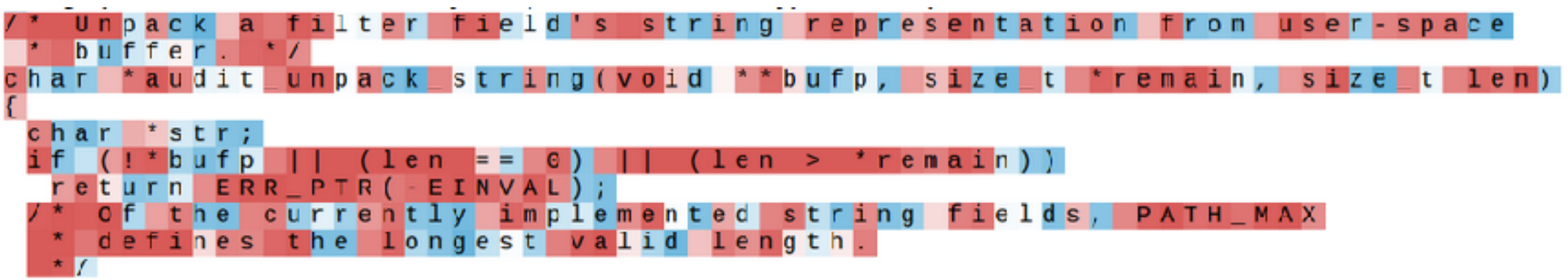
Inside/Outside IF

```
#ifdef CONFIG_AUDITSYSCALL
static inline int audit_match_class_bits(int class, u32 *mask)
{
  int i;
  if (classes[class]) {
    for (i = 0; i < AUDIT_BITMASK_SIZE; i++)
      if (mask[i] & classes[class][i])
        return 0;
  }
  return 1;
}
```

Depth of Expression

all hail andrej karpathy

Interpreting LSTM Cells



```
/*  Unpack a filter field's string representation from user-space
 *  buffer.  */
char *audit_unpack_string(void **bufp, size_t *remain, size_t len)
{
  char *str;
  if (!*bufp || (len == 0) || (len > *remain))
    return ERR_PTR(-EINVAL);
  /*  Of the currently implemented string fields, PATH_MAX
   *  defines the longest valid length.
   */
```

Uninterpretable

all hail andrej karpathy

“Deep” RNNs

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$h \in \mathbb{R}^n$ $\quad$ $W^l \ [n \times 2n]$

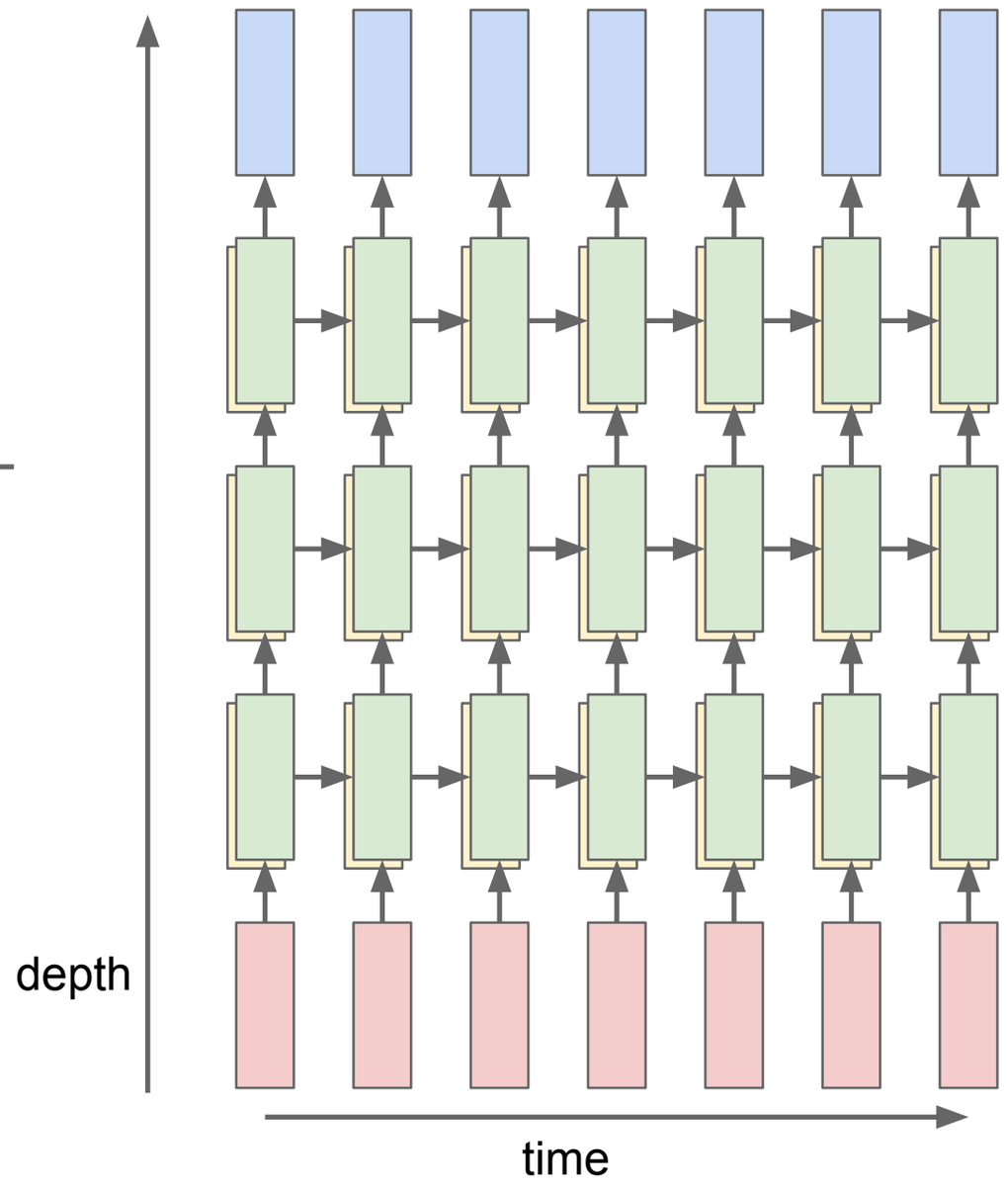
LSTM:

$W^l \ [4n \times 2n]$

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

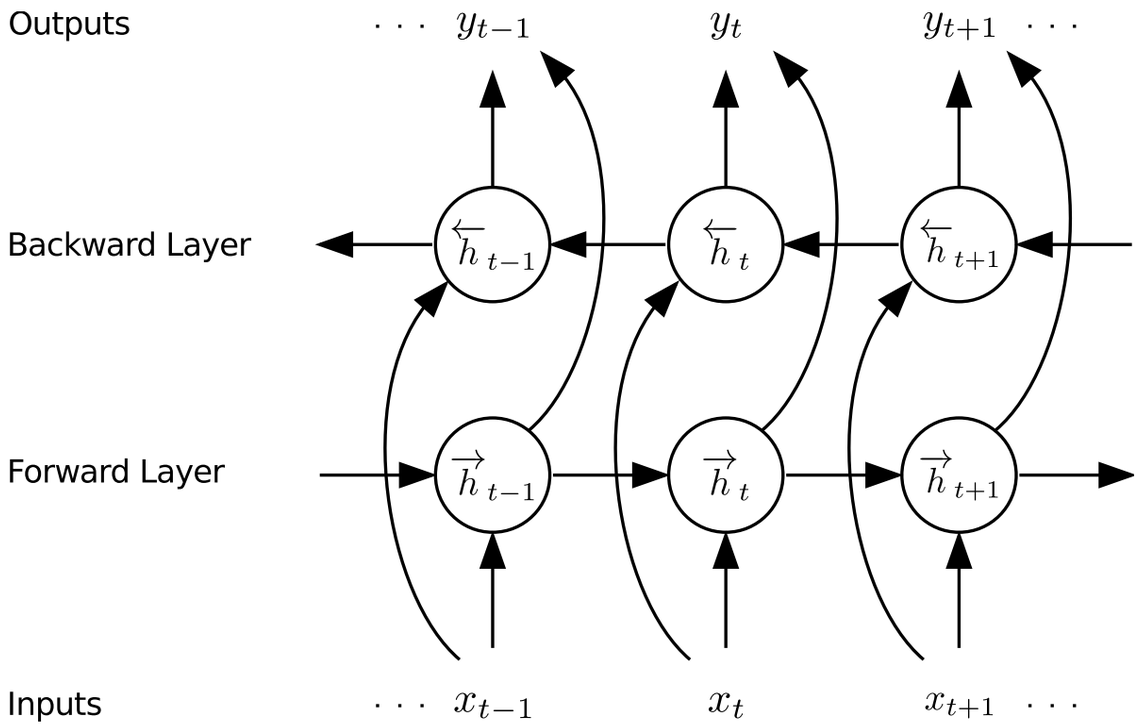


Note: notations are different

all hail andrej karpathy

Other Fancy Architectures

- BRNNs



$$\vec{h}_t = f(\overrightarrow{W}x_t + \overrightarrow{V}\vec{h}_{t-1} + \vec{b})$$

$$\overleftarrow{h}_t = f(\overleftarrow{W}x_t + \overleftarrow{V}\overleftarrow{h}_{t+1} + \overleftarrow{b})$$

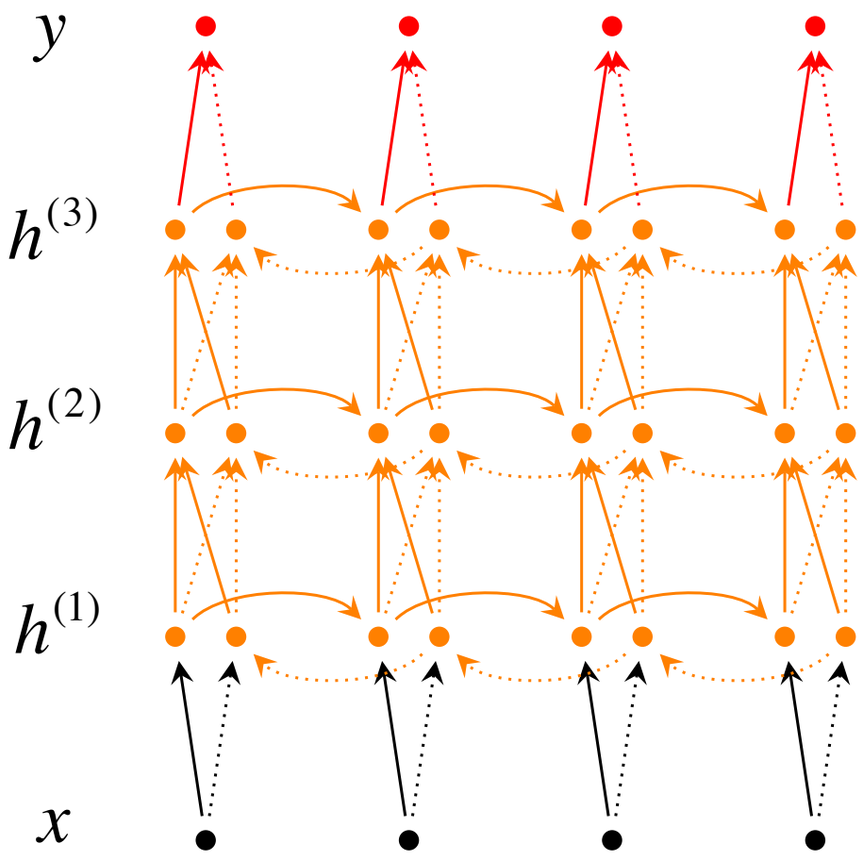
$$y_t = g(U[\vec{h}_t; \overleftarrow{h}_t] + c)$$

Note: notations are different

A. Graves and J. Schmidhuber, “Framewise Phoneme Classification with Bidirectional LSTM and Other Neural Network Architectures,” *Neural Networks*, June/July 2005.

Other Fancy Architectures

- Deep BRNNs



$$\overrightarrow{h}_t^{(i)} = f(\overrightarrow{W}^{(i)} h_t^{(i-1)} + \overrightarrow{V}^{(i)} \overrightarrow{h}_{t-1}^{(i)} + \overrightarrow{b}^{(i)})$$

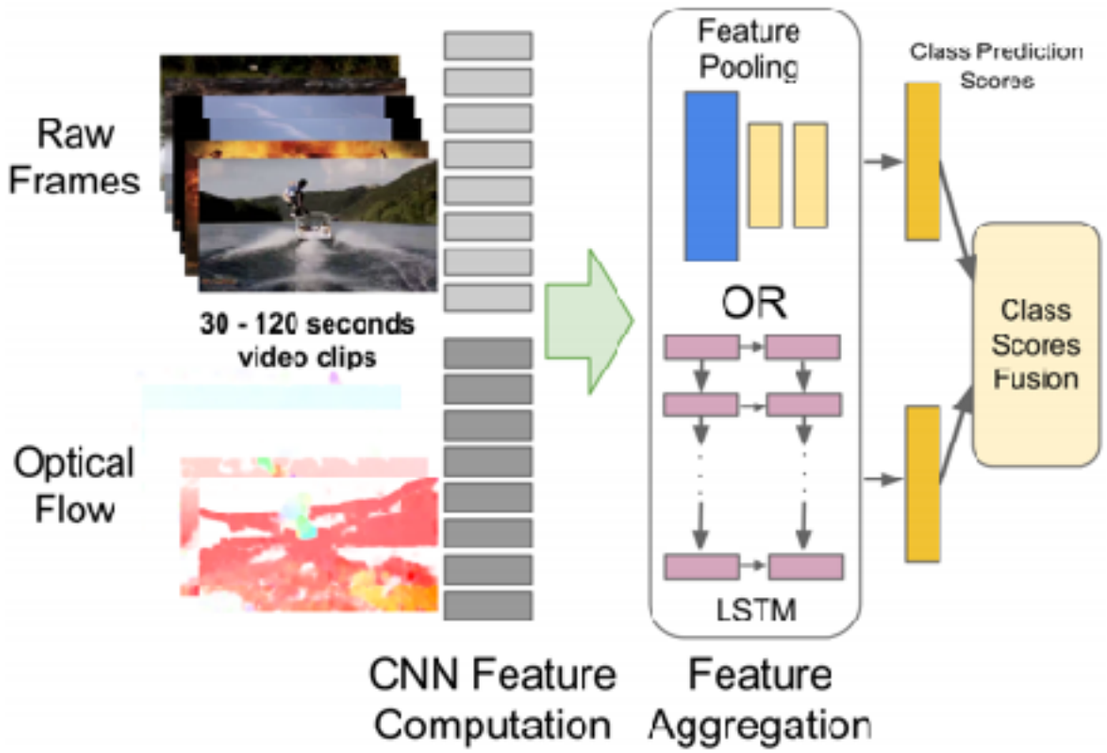
$$\overleftarrow{h}_t^{(i)} = f(\overleftarrow{W}^{(i)} h_t^{(i-1)} + \overleftarrow{V}^{(i)} \overleftarrow{h}_{t+1}^{(i)} + \overleftarrow{b}^{(i)})$$

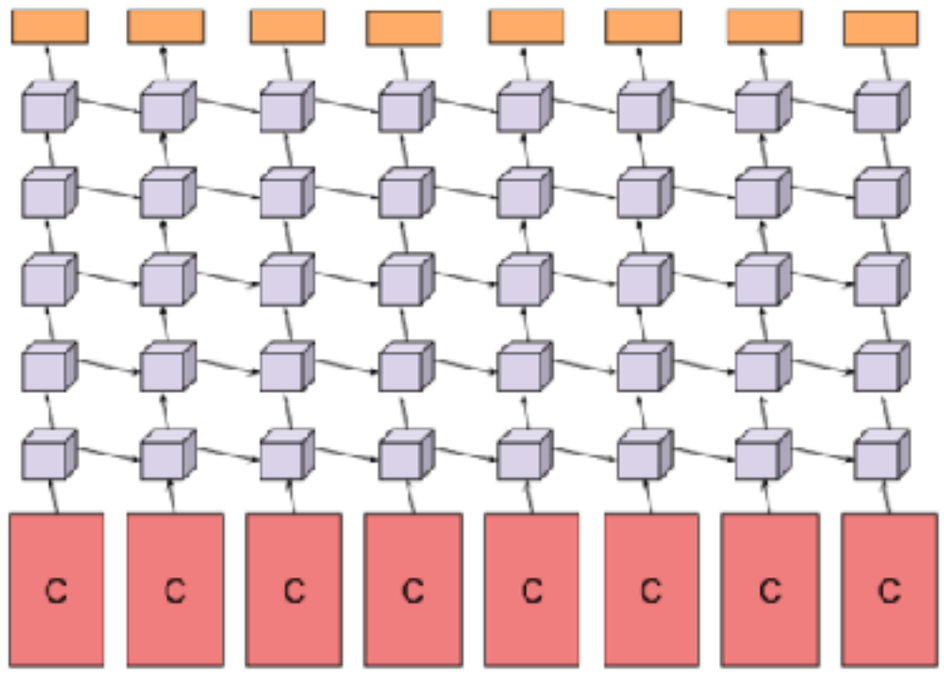
$$y_t = g(U[\overrightarrow{h}_t^{(L)}; \overleftarrow{h}_t^{(L)}] + c)$$

Note: notations are different

Picture courtesy: Richard Socher

Other Fancy Architectures





Ng et al., Beyond Short Snippets: Deep Networks for Video Classification, CVPR 2015

Next Session

"man in black shirt is playing guitar."

Image Captioning

Visual Question Answering

Summary

- RNNs & LSTMs are Awesome
  - Took us 20 years to realise it!
- Remember those LSTM equations
- Can you design your own architectures?
  - And Cells?

Online References

- http://karpathy.github.io/2015/05/21/rnn-effectiveness/
- https://colah.github.io/posts/2015-08-Understanding-LSTMs/
- http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/

References

LSTMs

- Sepp Hochreiter and Jurgen Schmidhuber. Long Short-Term Memory. Neural Computation, November 1997

- Gref et al., LSTM: A Search Space Odyssey, IEEE Trans. Neural Networks and Learning Systems, Oct. 2017.

- Pascanu et al., On the difficulty of training recurrent neural networks, ICML 2013

References

GRUs

- Cho et al., Learning Phrase Representations using RNN Encoder Decoder for Statistical Machine Translation. *arXiv:1406.1078, 2014*

RNNs

- Bengio et al., Advances in optimizing recurrent networks, IEEE ICASSP, 2013

CNNs for Video

- Simonyan & Zisserman, Two-Stream Convolutional Networks for Action Recognition in Videos, NIPS 2014

Thank You

Questions?